Guia Foca GNU/Linux

Gleydson Mazioli da Silva <gleydson@guiafoca.org>

Versão 4.22 - domingo, 05 de setembro de 2010

Resumo

Este documento tem por objetivo ser uma referência ao aprendizado do usuário e um guia de consulta, operação e configuração de sistemas Linux (e outros tipos de *ix). A última versão deste guia pode ser encontrada na Página Oficial do Foca GNU/Linux (http://www.guiafoca.org). Novas versões são lançadas com uma freqüência mensal e você pode receber avisos de novos lançamentos deste guia preenchendo um formulário na página Web.

Sumário

Capítulo 1

Introdução

Bem vindo ao guia *Foca GNU/Linux*. O nome *FOCA* significa *FO*nte de *C*onsulta e *A*prendizado. Este guia é dividido em 3 níveis de aprendizado e versão que esta lendo agora contém:

- Iniciante

Entre o conteúdo do guia, você encontrará:

- Textos explicativos falando sobre o sistema `Linux`, seus comandos, como manusear arquivos, diretórios, etc.
- Explicações iniciais sobre as partes básicas do computador e periféricos
- Comandos e Programas equivalentes entre o `DOS/Windows` e o `GNU/Linux`
- Todos os materiais contidos na versão iniciante são ideais para quem está tendo o primeiro contato com computadores e/ou com o `Linux`. A linguagem usada é simples com o objetivo de explicar claramente o funcionamento de cada comando e evitando, sempre que possível, termos técnicos

Para melhor organização, dividi o guia em 3 versões: *Iniciante*, *Intermediário* e *Avançado*. Sendo que a versão *Iniciante* é voltada para o usuário que não tem `nenhuma` experiência no `GNU/Linux`. A última versão deste guia pode ser encontrada em: Página Oficial do guia Foca GNU/Linux (`http://www.guiafoca.org`).

Caso tiver alguma sugestão, correção, crítica para a melhoria deste guia, envie um e-mail para `<gleydson@guiafoca.org>`.

O *Foca GNU/Linux* é atualizado freqüentemente, por este motivo recomendo que preencha a ficha do aviso de atualizações na página web em Página Oficial do guia Foca GNU/Linux (`http://www.guiafoca.org`) no fim da página principal. Após preencher a ficha do aviso de atualizações, você receberá um e-mail sobre o lançamento de novas versões do guia e o que foi modificado, desta forma você poderá decidir em copia-la caso a nova versão contenha modificações que considera importantes.

Tenho recebido elegios de pessoas do Brasil (e de paises de fora também) elogiando o trabalho e a qualidade da documentação. Agradeço a todos pelo apoio, tenham certeza que este trabalho é desenvolvido pensando em repassar um pouco do conhecimento que adquiri ao começar o uso do Linux.

Também venho recebendo muitos e-mails de pessoas que passaram na prova LPI nível 1 e 2 após estudar usando o guia Foca GNU/Linux. Fico bastante feliz por saber disso, pois nunca tive a intenção de tornar o guia uma referência livre para estudo da LPI e hoje é usado para estudo desta difícil certificação que aborda comandos, serviços, configurações, segurança, empacotamento, criptografia, etc.

1.1 Antes de começar

Os capítulos *Introdução* e *básico* contém explicações teóricas sobre o computador, `GNU/Linux`, etc., você pode pular este capítulos caso já conheça estas explicações ou se desejar partir para a prática e quiser vê-los mais tarde, se lhe interessar.

Se você já é um usuário do `DOS` e `Windows`, recomendo ler 'Para quem esta migrando (ou pensando em migrar) do DOS/Windows para o Linux' on page 35. Lá você vai encontrar comparações de comandos e programas `DOS/Windows` e `GNU/Linux`.

Para quem está começando, muita teoria pode atrapalhar o aprendizado, é mais produtivo ver na prática o que o computador faz e depois porque ele faz isto. Mesmo assim, recomendo ler estes capítulos pois seu conteúdo pode ser útil. . .

Coloquei abaixo algumas dicas para um bom começo:

- Recomendo que faça a leitura deste guia e pratique imediatamente o que aprendeu. Isto facilita o entendimento do programa/comando/configuração.
- É preciso ter interesse em aprender, se você tiver vontade em aprender algo, você terá menos dificuldade do que em algo que não gosta e está se obrigando a aprender.
- Decorar não adianta, pelo contrário, só atrapalha no aprendizado. Você precisa entender o que o comando faz, deste modo você estará estimulando e desenvolvendo sua interpretação, e entenderá melhor o assunto (talvez até me de uma força para melhorar o guia ;-)
- Curiosidade também é importante. Você talvez possa estar procurando um comando que mostre os arquivos que contém um certo texto, e isto fará você chegar até o comando `grep`, depois você conhecerá suas opções, etc.
- Não desanime vendo outras pessoas que sabem mais que você, lembre-se que ninguém nasce sabendo :-). Uma pessoa pode ter mais experiência em um assunto no sistema como compilação de programas, configuração, etc., e você pode ter mais interesse em redes.
- Ninguém pode saber tudo da noite para o dia, não procure saber TUDO sobre o sistema de uma só vez, senão não entenderá NADA. Caso tenha dúvidas sobre o sistema, procure ler novamente a seção do guia, e caso ainda não tenha entendido procure ajuda nas página de manual (veja 'Páginas de Manual' on page 119), ou nas listas de discussão (veja 'Listas de discussão' on page 125) ou me envie uma mensagem `<gleydson@guiafoca.org>`.

- Certamente você buscará documentos na Internet que falem sobre algum assunto que este guia ainda não explica. Muito cuidado! O GNU/Linux é um sistema que cresce muito rapidamente, a cada semana uma nova versão é lançada, novos recursos são adicionados, seria maravilhoso se a documentação fosse atualizada com a mesma freqüência. Infelizmente a atualização da documentação não segue o mesmo ritmo (principalmente aqui no Brasil). É comum você encontrar na Internet documentos da época quando o kernel estava na versão 2.2.30, 2.4.8, 2.6.28, etc. Estes documentos são úteis para pessoas que por algum motivo necessitam operar com versões antigas do Kernel Linux, mas pode trazer problemas ou causar má impressão do GNU/Linux em outras pessoas. Por exemplo, você pode esbarrar pela Internet com um documento que diz que o Kernel não tem suporte aos "nomes extensos" da VFAT (Windows 95), isto é verdade para kernels anteriores ao 2.0.31, mas as versões mais novas que a 2.0.31 reconhecem sem problemas os nomes extensos da partição Windows VFAT. Uma pessoa desavisada pode ter receio de instalar o GNU/Linux em uma mesma máquina com Windows por causa de um documento como este. Para evitar problemas deste tipo, verifique a data de atualização do documento, se verificar que o documento está obsoleto, contacte o autor original e peça para que ele retire aquela seção na próxima versão que será lançada.
- O GNU/Linux é considerado um sistema mais difícil do que os outros, mas isto é porque ele requer que a pessoa realmente aprenda e conheça computadores e seus periféricos antes de fazer qualquer coisa (principalmente se você é um técnico em manutenção, redes, instalações, etc., e deseja oferecer suporte profissional a este sistema). Você conhecerá mais sobre computadores, redes, hardware, software, discos, saberá avaliar os problemas e a buscar a melhor solução, enfim as possibilidades de crescimento neste sistema operacional depende do conhecimento, interesse e capacidade de cada um.
- A interface gráfica existe, mas os melhores recursos e flexibilidade estão na linha de comando. Você pode ter certeza que o aprendizado no GNU/Linux ajudará a ter sucesso e menos dificuldade em usar qualquer outro sistema operacional.
- Peça ajuda a outros usuários do GNU/Linux quando estiver em dúvida ou não souber fazer alguma coisa no sistema. Você pode entrar em contato diretamente com outros usuários ou através de listas de discussão (veja 'Listas de discussão' on page 125).

Boa Sorte e bem vindo ao GNU/Linux!

Gleydson (<gleydson@guiafoca.org>).

1.2 Pré-requisitos para a utilização deste guia

É assumido que você já tenha seu GNU/Linux instalado e funcionando.

Este guia não cobre a instalação do sistema. Para detalhes sobre instalação, consulte a documentação que acompanha sua distribuição GNU/Linux.

1.3 Sistema Operacional

O *Sistema Operacional* é o conjunto de programas que fazem a interface do usuário e seus programas com o computador. Ele é responsável pelo gerenciamento de recursos e periféricos (como memória, discos, arquivos, impressoras, CD-ROMs, etc.), interpretação de mensagens e a execução de programas.

No `Linux` o Kernel mais o conjunto de ferramentas GNU compõem o Sistema Operacional. O kernel (que é a base principal de um sistema operacional), poderá ser construído de acordo com a configuração do seu computador e dos periféricos que possui.

1.4 O Linux

O `Linux` é um sistema operacional criado em 1991 por *Linus Torvalds* na universidade de Helsinki na Finlândia. É um sistema Operacional de código aberto distribuído gratuitamente pela Internet. Seu código fonte é liberado como *Free Software* (software livre), sob licença GPL, o aviso de copyright do kernel feito por Linus descreve detalhadamente isto e mesmo ele não pode fechar o sistema para que seja usado apenas comercialmente.

Isto quer dizer que você não precisa pagar nada para usar o Linux, e não é crime fazer cópias para instalar em outros computadores, nós inclusive incentivamos você a fazer isto. Ser um sistema de código aberto pode explicar a performance, estabilidade e velocidade em que novos recursos são adicionados ao sistema.

O requisito mínimo para rodar o `Linux` depende do kernel que será usado:
- `2.2.x` - Computador 386 SX com 2 MB de memória
- `2.4.x` - Computador 386 SX com 4MB de memória
- `2.6.x` - Computador 486 DX com no mínimo 8MB

Para espaço em disco é requerido 500MB para uma instalação básica usando modo texto com suporte a rede. Claro que não é considerada a execução de ambiente gráfico ou serviços de rede em produção, que neste caso é exigido mais memória RAM e espaço em disco para armazenamento de dados de programas e usuários.

O sistema segue o padrão *POSIX* que é o mesmo usado por sistemas *UNIX* e suas variantes. Assim, aprendendo o `Linux` você não encontrará muita dificuldade em operar um sistema do tipo `UNIX, FreeBSD, HPUX, SunOS,` etc., bastando apenas aprender alguns detalhes encontrados em cada sistema.

O código fonte aberto permite que qualquer pessoa veja como o sistema funciona (útil para aprendizado), corrigir algum problema ou fazer alguma sugestão sobre sua melhoria, esse é um dos motivos de seu rápido crescimento, do aumento da compatibilidade de periféricos (como novas placas sendo suportadas logo após seu lançamento) e de sua estabilidade.

Outro ponto em que ele se destaca é o suporte que oferece a placas, CD/DVD-RWs, BluRay e outros tipos de dispositivos de última geração e mais antigos (a maioria deles já ultrapassados e sendo completamente suportados pelo sistema operacional). Este é um ponto forte para

empresas que desejam manter seus micros em funcionamento e pretendem investir em avanços tecnológicos com as máquinas que possui.

O Linux é desenvolvido por milhares de pessoas espalhadas pelo mundo, cada uma fazendo sua contribuição ou mantendo alguma parte do kernel gratuitamente. *Linus Torvalds* ainda trabalha em seu desenvolvimento e na coordenação dos grupos de trabalho do kernel.

O suporte ao sistema também se destaca como sendo o mais eficiente e rápido do que qualquer programa comercial disponível no mercado. Existem milhares de consultores e empresas especializadas no suporte e treinamento espalhados ao redor do mundo. Outra opção de suporte é através da comunidade Linux; você pode se inscrever em uma lista de discussão e relatar sua dúvida ou alguma falha, e sua mensagem será vista por centenas de usuários na Internet e algum irá te ajudar ou avisará as pessoas responsáveis sobre a falha encontrada para devida correção. Para detalhes, veja 'Listas de discussão' on page 125.

1.4.1 Algumas Características do Linux

- É livre e desenvolvido voluntariamente por programadores experientes, hackers, e contribuidores espalhados ao redor do mundo que tem como objetivo a contribuição para a melhoria e crescimento deste sistema operacional. Muitos deles estavam cansados do excesso de propaganda (Marketing) e baixa qualidade de sistemas comerciais existentes
- Também recebe apoio de grandes empresas como IBM, Sun, HP, etc. para seu desenvolvimento
- Convivem sem nenhum tipo de conflito com outros sistemas operacionais (com o DOS, Windows, OS/2) no mesmo computador.
- Multitarefa real
- Multiusuário
- Suporte a nomes extensos de arquivos e diretórios (255 caracteres)
- Conectividade com outros tipos de plataformas como *Apple, Sun, Macintosh, Sparc, Alpha, PowerPc, ARM, Unix, Windows, DOS, etc.*
- Utiliza permissões de acesso a arquivos, diretórios e programas em execução na memória RAM.
- Proteção entre processos executados na memória RAM
- Suporte a mais de 63 terminais virtuais (consoles)
- Modularização - O Linux somente carrega para a memória o que é usado durante o processamento, liberando totalmente a memória assim que o programa/dispositivo é finalizado
- Devido a modularização, os drivers dos periféricos e recursos do sistema podem ser carregados e removidos completamente da memória RAM a qualquer momento. Os drivers (módulos) ocupam pouco espaço quando carregados na memória RAM (cerca de 6Kb para a Placa de rede NE 2000, por exemplo)
- Suporte nativo a rede e tecnologias avançadas como: balanceamento de carga, ips alias, failover, vlans, bridge, trunking, OSPF, BGP.
- Não há a necessidade de se reiniciar o sistema após a modificar a configuração de qualquer periférico ou parâmetros de rede. Somente é necessário reiniciar o sistema no caso de uma instalação interna de um novo periférico, falha em algum hardware (queima do processador, placa mãe, etc.).

- Não precisa de um processador potente para funcionar. O sistema roda bem em computadores 386Sx 25 com 4MB de memória RAM (sem rodar o sistema gráfico X, que é recomendado 32MB de RAM). Já pensou no seu desempenho em um Pentium, Xeon, ou Athlon? ;-)
- Suporte nativo a múltiplas CPUs, assim processadores como Dual Core, Core Duo, Athlon Duo, Quad Core tem seu poder de processamento integralmente aproveitado, tanto em 32 ou 64 bits.
- Suporte nativo a dispositivos SATA, PATA, Fiber Channel
- Suporte nativo a virtualização, onde o `Linux` se destaca como plataforma preferida para execução de múltiplos sistemas operacionais com performance e segurança.
- O crescimento e novas versões do sistema não provocam lentidão, pelo contrário, a cada nova versão os desenvolvedores procuram buscar maior compatibilidade, acrescentar recursos úteis e melhor desempenho do sistema (como o que aconteceu na passagem do kernel 2.0.x para 2.2.x, da 2.2.x para a 2.4.x e da 2.4.x para a 2.6.x).
- O `GNU/Linux` é distribuido livremente e licenciado de acordo com os termos da GPL.
- Acessa corretamente discos formatados pelo `DOS`, `Windows`, `Novell`, `OS/2`, `NTFS`, `SunOS`, `Amiga`, `Atari`, `Mac`, etc.
- O LINUX NÃO É VULNERÁVEL A VÍRUS! Devido a separação de privilégios entre processos e respeitadas as recomendações padrão de política de segurança e uso de contas privilegiadas (como a de root, como veremos adiante), programas como vírus tornam-se inúteis pois tem sua ação limitada pelas restrições de acesso do sistema de arquivos e execução. Qualquer programa (nocivo ou não) poderá alterar partes do sistema que possui permissões (será abordado como alterar permissões e tornar seu sistema mais restrito no decorrer do guia). Frequentemente são criados exploits que tentam se aproveitar de falhas existentes em sistemas desatualizados e usa-las para causar danos. *Erroneamente* este tipo de ataque é classificado como vírus por pessoas mal informadas e são resolvidas com sistemas bem mantidos. Em geral, usando uma boa distribuição que tenha um eficiente sistema de atualização e bem configurado, você terá 99.9% de sua tranquilidade.
- Rede TCP/IP mais rápida que no Windows e tem sua pilha constantemente melhorada. O `GNU/Linux` tem suporte nativo a redes TCP/IP e não depende de uma camada intermediária como o WinSock. Em acessos via modem a Internet, a velocidade de transmissão é 10% maior.
- Executa outros sistemas operacionais como *Windows*, *MacOS*, *DOS* ou outro sistema `Linux` através de consagrados sistemas de virtualização como `Xen`, `vmware`, ou emulação como o `DOSEMU`, `QEMU`, `WINE`.
- Suporte completo e nativo a diversos dispositivos de comunicação via infravermelho, Bluetooth, Firewire, USB. Basta conectar e o seu dispositivo é automaticamente reconhecido. Raramente são necessários drivers externos, exceto no caso de dispositivos muito novos que não tenham o suporte ainda adicionado no sistema.
- Suporte a rede via rádio amador.
- Suporte a dispositivos Plug-and-Play.
- Suporte nativo a pen drivers, dispositivos de armazenamento e cartões de memória.
- Suporte nativo a dispositivos I2C
- Integração com gerenciamento de energia ACPI e APM
- Dispositivos de rede Wireless. Tanto com criptografia WEB e WPA PSK
- Vários tipos de firewalls avançados de alta qualidade na detecção de tráfego indesejável,

dando ao administrador uma excelente ferramenta de proteção e controle de sua rede.
- Roteamento estático e dinâmico de pacotes.
- Ponte entre Redes, proxy arp
- Proxy Tradicional e Transparente.
- Possui recursos para atender a mais de um endereço IP na mesma placa de rede, sendo muito útil para situações de manutenção em servidores de redes ou para a emulação de "múltiplos computadores". O servidor WEB e FTP podem estar localizados no mesmo computador, mas o usuário que se conecta tem a impressão que a rede possui servidores diferentes.
- Os sistemas de arquivos usados pelo `GNU/Linux` (`Ext2`, `Ext3`, `reiserfs`, `xfs`, `jfs`) organiza os arquivos de forma inteligente evitando a fragmentação e fazendo-o um poderoso sistema para aplicações multi-usuárias exigentes e gravações intensivas.
- Permite a montagem de um servidor de publicação Web, E-mail, News, etc. com um baixo custo e alta performance. O melhor servidor Web do mercado, o `Apache`, é distribuído gratuitamente junto com a maioria das distribuições Linux. O mesmo acontece com o `Sendmail`.
- Por ser um sistema operacional de código aberto, você pode ver o que o código fonte (instruções digitadadas pelo programador) faz e adapta-lo as suas necessidades ou de sua empresa. Esta característica é uma segurança a mais para empresas sérias e outros que não querem ter seus dados roubados (você não sabe o que um sistema sem código fonte faz na realidade enquanto esta processando o programa).
- Suporte a diversos dispositivos e periféricos disponíveis no mercado, tanto os novos como obsoletos.
- Pode ser executado em 16 arquiteturas diferentes (Intel, Macintosh, Alpha, Arm, etc.) e diversas outras sub-arquiteturas.
- Empresas especializadas e consultores especializados no suporte ao sistema espalhados por todo o mundo.
- Entre muitas outras características que você descobrirá durante o uso do sistema (além de poder criar outras, caso seja um administrador avançado ou desenvolvedor).

TODOS OS ÍTENS DESCRITOS ACIMA SÃO VERDADEIROS E TESTADOS PARA QUE TIVESSE PLENA CERTEZA DE SEU FUNCIONAMENTO.

1.5 Distribuições do Linux

Só o kernel `GNU/Linux` não é suficiente para se ter uma sistema funcional, mas é o principal.

Existem grupos de pessoas, empresas e organizações que decidem "distribuir" o Linux junto com outros aplicativos (como por exemplo `editores gráficos, planilhas, bancos de dados, ambientes de programação, formatação de documentos, firewalls, etc`).

Este é o significado essencial de *distribuição*. Cada distribuição tem sua característica própria, como o sistema de instalação, o objetivo, a localização de programas, nomes de arquivos de configuração, etc. A escolha de uma distribuição é pessoal e depende das necessidades de cada um.

Algumas distribuições bastante conhecidas são: *Ubuntu, Debian, Slackware, Red Hat, Gentoo, Suse* todas usando o SO Linux como kernel principal (a `Debian` é uma distribuição independente de kernel e pode ser executada sob outros kernels, como o GNU hurd ou o kernel BSD).

A escolha de sua distribuição deve ser feita com muita atenção, não adianta muita coisa perguntar em canais de IRC sobre qual é a melhor distribuição, ser levado pelas propagandas, pelo vizinho, etc. O melhor caminho para a escolha da distribuição, acredito eu, seria perguntar as características de cada uma e porque essa pessoa gosta dela ao invés de perguntar qual é a melhor, porque quem lhe responder isto estará usando uma distribuição que se encaixa de acordo com suas necessidade e esta mesma distribuição pode não ser a melhor para lhe atender.

Segue abaixo as características de algumas distribuições seguidas do site principal e endereço para download:

Debian `http://www.debian.org/` - Distribuição desenvolvida e atualizada através do esforço de voluntários espalhados ao redor do mundo, seguindo o estilo de desenvolvimento `GNU/Linux`. Por este motivo, foi adotada como a distribuição oficial do projeto *GNU*. Possui suporte a língua Portuguesa, é a única que tem suporte a 14 arquiteturas diferentes (i386, IA64, AMD64, Alpha, Sparc, PowerPc, Macintosh, Arm, etc.) e aproximadamente 15 sub-arquiteturas. A instalação da distribuição pode ser feita tanto através de Disquetes, CD-ROM, Tftp, Ftp, NFS ou através da combinação de vários destes em cada etapa de instalação.

Acompanha mais de 25000 programas distribuídos em forma de pacotes cada um destes programas são mantidos e testados pela pessoa ou grupo responsável por seu empacotamento. Os pacotes são divididos em diretórios de acordo com sua categoria e gerenciados através de um avançado sistema de gerenciamento de pacotes (o apt e o dpkg) facilitando a instalação e atualização de pacotes. Possui tanto ferramentas para administração de redes e servidores quanto para desktops, estações multimídia, jogos, desenvolvimento, web, etc.

A atualização da distribuição ou de pacotes individuais pode ser feita facilmente através de 2 comandos, não requerendo adquirir um novo CD para usar a última versão da distribuição. É a única distribuição não comercial onde todos podem contribuir usando seu conhecimento para o desenvolvimento. Para gerenciar os voluntários, conta com centenas de listas de discussão envolvendo determinados desenvolvedores das mais diversas partes do mundo.

São feitos extensivos testes antes do lançamento de cada versão para atingir um alto grau de confiabilidade. As falhas encontradas nos pacotes podem ser relatados através de um *sistema de tratamento de falhas* que encaminha a falha encontrada diretamente ao responsável para avaliação e correção. Qualquer um pode receber a lista de falhas ou sugestões sobre a distribuição cadastrando-se em uma das lista de discussão que tratam especificamente da solução de falhas encontradas na distribuição (disponível na página principal da distribuição).

Os pacotes podem ser instalados através de `Tarefas` contendo seleções de pacotes de acordo com a utilização do computador (servidor Web, desenvolvimento, TeX, jogos,

desktop, etc.), *Perfis* contendo seleções de pacotes de acordo com o tipo de usuário (programador, operador, etc.), ou através de uma seleção individual de pacotes, garantindo que somente os pacotes selecionados serão instalados fazendo uma instalação enxuta.

Existe um time de desenvolvedores com a tarefa específica de monitorar atualizações de segurança em serviços (apache, sendmail, e todos os outros 25000 pacotes) que possam comprometer o servidor, deixando-o vulnerável a ataques. Assim que uma falha é descoberta, é enviado uma alerta (DSA - Debian Security Alert) e disponibilizada uma atualização para correção das diversas versões da `Debian`. Isto é geralmente feito em menos de 48 horas desde a descoberta da falha até a divulgação da correção. Como quase todas as falhas são descobertas nos programas, este método também pode ser usado por administradores de outras distribuições para manterem seu sistema seguro e atualizado.

O suporte ao usuário e desenvolvimento da distribuição são feitos através de listas de discussões e canais IRC. Existem uma lista de consultores habilitados a dar suporte e assistência a sistemas Debian ao redor do mundo na área consultores do site principal da distribuição.

ftp://ftp.debian.org/ - Endereço para download.

Ubuntu http://www.ubuntu.com/ - Variante da distribuição Debian voltada a interação mais amigável com o usuário final e facilidade de instalação. Atualmente é a melhor para usuários que tem o primeiro contato com o Linux. Conta tanto com a instalação do sistema em HD e execução através de Live CD.

http://www.ubuntu.com/getubuntu/download/ - Endereço para download do Ubuntu.

Slackware http://www.slackware.com/ - Distribuição desenvolvida por `Patrick Volkerding`, desenvolvida para alcançar facilidade de uso e estabilidade como prioridades principais. Foi a primeira distribuição a ser lançada no mundo e costuma trazer o que há de mais novo enquanto mantém uma certa tradição, provendo simplicidade, facilidade de uso e com isso flexibilidade e poder.

Desde a primeira versão lançada em Abril de 1993, o Projeto `Slackware Linux` tem buscado produzir a distribuição `Linux` mais `UNIX-like`, ou seja, mais parecida com UNIX. O Slackware segue os padrões Linux como o Linux File System Standard, que é um padrão de organização de diretórios e arquivos para as distribuições.

Enquanto as pessoas diziam que a Red Hat era a melhor distribuição para o usuário iniciante, o `Slackware` é o melhor para o usuário mais "velho", ou seja programadores, administradores, etc.

ftp://ftp.slackwarebrasil.org/linux/slackware/ - Ftp da distribuição Slackware.

SuSE http://www.suse.com/ - Distribuição comercial Alemã com a coordenação sendo feita através dos processos administrativos dos desenvolvedores e de seu braço norte-americano. O foco da Suse é o usuário com conhecimento técnico no Linux (programador, administrador de rede, etc.) e não o usuário iniciante no Linux. Preferencialmente a administração deve ser feita usando o `Yast`, mas também pode ser feita manualmente através de alteração dos arquivos de configuração.

Possui suporte as arquiteturas Intel x86 e Alpha. Sua instalação pode ser feita via CD-ROM ou CD-DVD (é a primeira distribuição com instalação através de DVD).

Uma média de 2000 programas acompanham a versão 10 distribuídos em 6 CD-ROMs. O sistema de gerenciamento de pacotes é o RPM padronizado. A seleção de pacotes durante a instalação pode ser feita através da seleção do perfil de máquina (developer, estação kde, gráficos, estação gnome, servidor de rede, etc.) ou através da seleção individual de pacotes.

A atualização da distribuição pode ser feita através do CD-ROM de uma nova versão ou baixando pacotes de ftp://ftp.suse.com/. Usuários registrados ganham direito a suporte de instalação via e-mail. A base de dados de suporte também é excelente e está disponível na web para qualquer usuário independente de registro.

ftp://ftp.suse.com/ - Ftp da distribuição SuSE.

Red Hat Enterprise Linux http://www.redhat.com/ - Distribuição comercial suportada pela Red Hat e voltada a servidores de grandes e medias empresas. Também conta com uma certificação chamada RHCE específica desta distro.

Ela não está disponível para download, apenas vendida a custos a partir de 179 dólares (a versão workstation) até 1499 dólares (advanced server).

Fedora http://fedora.redhat.com/ - O Fedora Linux é a distribuição de desenvolvimento aberto patrocinada pela RedHat e pela comunidade, originada em 2002 e baseada em versão da antiga linha de produtos RedHat Linux. Esta distribuição não é suportada pela Red Hat como distribuição oficial (ela suporta apenas a linha Red Hat Enterprise Linux), devendo obter suporte através da comunidade ou outros meios.

A distribuição `Fedora` dá prioridade ao uso do computador como estação de trabalho. Além de contar com uma ampla gama de ferramentas de escritório possui funções de servidor e aplicativos para produtividade e desenvolvimento de softwares. Considerado um dos sistemas mais fáceis de instalar e utilizar, inclui tradução para portugês do Brasil e suporte às plataformas Intel e 64 bits.

Por basear-se no RedHat. o Fedora conta com um o `up2date`, um software para manter o sistema atualizado e utiliza pacotes de programas no formato RPM, um dos mais comuns.

O Fedora não é distribuido oficialmente através de mídias ou CDs, se você quiser obte-lo terá de procurar distribuidores independentes ou fazer o download dos 4 CDs através do site oficial.

http://download.fedora.redhat.com/pub/fedora/linux/core/2/i386/iso/ - Download da distribuição Fedora.

Mandriva http://www.mandriva.com/ - Fusão da distribuição francesa `Mandrake` com a distribuição brasileira `Conectiva` contendo as características de instalação semi-automática através de DVD. Boa auto-detecção de periféricos, inclusive web-cams.

http://www.mandriva.com/ - Download da distribuição.

Para contato com os grupos de usuários que utilizam estas distribuições, veja 'Listas de discussão' on page 125.

1.6 Software Livre

(tradução do texto `Linux e o Sistema GNU` de `Richard Stallman` obtido no site do CIPSGA: `http://www.cipsga.org.br/`). O projeto *GNU* começou em 1983 com o objetivo de desenvolver um sistema operacional Unix-like totalmente livre. `Livre` se refere à liberdade, e não ao preço; significa que você está livre para executar, distribuir, estudar, mudar e melhorar o software.

Um sistema Unix-like consiste de muitos programas diferentes. Nós achamos alguns componentes já disponíveis como softwares livres – por exemplo, `X Window` e `TeX`. Obtemos outros componentes ajudando a convencer seus desenvolvedores a tornarem eles livres – por exemplo, o Berkeley network utilities. Outros componentes nós escrevemos especificamente para o GNU – por exemplo, `GNU Emacs`, o compilador `GNU C`, o `GNU C library`, `Bash` e `Ghostscript`. Os componentes desta última categoria são "software GNU". O sistema GNU consiste de todas as três categorias reunidas.

O projeto GNU não é somente desenvolvimento e distribuição de alguns softwares livres úteis. O coração do projeto GNU é uma idéia: que software deve ser *livre*, e que a liberdade do usuário vale a pena ser defendida. Se as pessoas têm liberdade mas não a apreciam conscientemente, não irão mantê-la por muito tempo. Se queremos que a liberdade dure, precisamos chamar a atenção das pessoas para a liberdade que elas têm em programas livres.

O método do projeto GNU é que programas livres e a idéia da liberdade dos usuários ajudam-se mutuamente. Nós desenvolvemos software GNU, e conforme as pessoas encontrem programas GNU ou o sistema GNU e comecem a usá-los, elas também pensam sobre a filosofia GNU. O software mostra que a idéia funciona na prática. Algumas destas pessoas acabam concordando com a idéia, e então escrevem mais programas livres. Então, o software carrega a idéia, dissemina a idéia e cresce da idéia.

Em 1992, nós encontramos ou criamos todos os componentes principais do sistema exceto o kernel, que nós estávamos escrevendo. (Este kernel consiste do microkernel Mach mais o GNU HURD. Atualmente ele está funcionando, mas não está preparado para os usuários. Uma versão alfa deverá estar pronta em breve.)

Então o kernel do Linux tornou-se disponível. Linux é um kernel livre escrito por Linus Torvalds compatível com o Unix. Ele não foi escrito para o projeto GNU, mas o Linux e o quase completo sistema GNU fizeram uma combinação útil. Esta combinação disponibilizou todos os principais componentes de um sistema operacional compatível com o Unix, e, com algum trabalho, as pessoas o tornaram um sistema funcional. Foi um sistema GNU variante, baseado no kernel do `Linux`.

Ironicamente, a popularidade destes sistemas desmerece nosso método de comunicar a idéia GNU para as pessoas que usam GNU. Estes sistemas são praticamente iguais ao sistema GNU – a principal diferença é a escolha do kernel. Porém as pessoas normalmente os chamam de "sistemas Linux (Linux systems)". A primeira impressão que se tem é a de que um "sistema Linux" soa como algo completamente diferente de "sistema GNU", e é isto que a maioria dos usuários pensam que acontece.

A maioria das introduções para o "sistema Linux" reconhece o papel desempenhado pelos

componentes de software GNU. Mas elas não dizem que o sistema como um todo é uma variante do sistema GNU que o projeto GNU vem compondo por uma década. Elas não dizem que o objetivo de um sistema Unix-like livre como este veio do projeto GNU. Daí a maioria dos usuários não saber estas coisas.

Como os seres humanos tendem a corrigir as suas primeiras impressões menos do que as informações subseqüentes tentam dizer-lhes, estes usuários que depois aprendem sobre a relação entre estes sistemas e o projeto GNU ainda geralmente o subestima.

Isto faz com que muitos usuários se identifiquem como uma comunidade separada de "usuários de Linux", distinta da comunidade de usuários GNU. Eles usam todos os softwares GNU; de fato, eles usam quase todo o sistema GNU; mas eles não pensam neles como usuários GNU, e freqüentemente não pensam que a filosofia GNU está relacionada a eles.

Isto leva a outros problemas também – mesmo dificultando cooperação com a manutenção de programas. Normalmente quando usuários mudam um programa GNU para fazer ele funcionar melhor em um sistema específico, eles mandam a mudança para o mantenedor do programa; então eles trabalham com o mantenedor explicando a mudança, perguntando por ela, e às vezes reescrevendo-a para manter a coerência e mantenebilidade do pacote, para ter o patch instalado.

Mas as pessoas que pensam nelas como "usuários Linux" tendem a lançar uma versão "Linux-only" do programa GNU, e consideram o trabalho terminado. Nós queremos cada e todos os programas GNU que funcionem "out of the box" em sistemas baseados em Linux; mas se os usuários não ajudarem, este objetivo se torna muito mais difícil de atingir.

Como deve o projeto GNU lidar com este problema? O que nós devemos fazer agora para disseminar a idéia de que a liberdade para os usuários de computador é importante?

Nós devemos continuar a falar sobre a liberdade de compartilhar e modificar software – e ensinar outros usuários o valor destas liberdades. Se nós nos beneficiamos por ter um sistema operacional livre, faz sentido para nós pensar em preservar estas liberdades por um longo tempo. Se nós nos beneficiamos por ter uma variedade de software livres, faz sentido pensar sobre encorajar outras pessoas a escrever mais software livre, em vez de software proprietário.

Nós não devemos aceitar a idéia de duas comunidades separadas para GNU e Linux. Ao contrário, devemos disseminar o entendimento de que "sistemas Linux" são variantes do sistema GNU, e que os usuários destes sistemas são tanto usuários GNU como usuários Linux (usuários do kernel do Linux). Usuários que têm conhecimento disto irão naturalmente dar uma olhada na filosofia GNU que fez estes sistemas existirem.

Eu escrevi este artigo como um meio de fazer isto. Outra maneira é usar os termos "sistema GNU baseado em Linux (Linux-based GNU system)" ou "sistema GNU/Linux (GNU/Linux system)", em vez de "sistema Linux", quando você escreve sobre ou menciona este sistema.

1.7 Processamento de Dados

Processamento de Dados é o envio de dados ao computador que serão processados e terão um resultado de saída útil.

Veja também 'Dispositivos de Entrada e Saída' on page 19.

1.8 O Computador

É uma máquina eletrônica que processa e armazena os dados e pode executar diversos programas para realizar uma série de tarefas e assim atender a necessidade do seu utilizador. O computador não é uma máquina inteligente, ele apenas executa as instruções dos programas que foram escritos pelo programador.

1.9 Conhecendo o Computador

Esta explica para que serve cada botão do painel do computador e monitor de vídeo. Se você já sabe para que cada um serve, recomendo pular esta parte, é o BE-A-BA. :-)

Todo computador possuem funções que são usados em outros tipos e modelos. Você pode ter um modelo de computador e um amigo seu outro tipo e mesmo tendo aparência diferente, terão as mesmas funções.

1.9.1 Tipos de Gabinete

Quanto ao tipo, o gabinete pode ser *Desktop*, *Mini-torre* e *Torre*.

Desktop É usado na posição *Horizontal* (como o vídeo cassete). Sua característica é que ocupa pouco espaço em uma mesa, pois pode ser colocado sob o monitor. A desvantagem é que normalmente possui pouco espaço para a colocação de novas placas e periféricos. Outra desvantagem é a dificuldade na manutenção deste tipo de equipamento (hardware).

Mini-Torre É usado na posição *Vertical* (torre). É o modelo mais usado. Sua característica é o espaço interno para expansão e manipulação de periféricos. A desvantagem é o espaço ocupado em sua mesa :-).

Torre Possui as mesmas características do *Mini-torre*, mas tem uma altura maior e mais espaço para colocação de novos periféricos. Muito usado em servidores de rede e placas que requerem uma melhor refrigeração.

1.9.2 Painel Frontal

O painel frontal do computador tem os botões que usamos para ligar, desligar, e acompanhar o funcionamento do computador. Abaixo o significado de cada um:

Botão POWER Liga/Desliga o computador.

Botão TURBO Se ligado, coloca a placa mãe em operação na velocidade máxima (o padrão). Desligado, faz o computador funcionar mais lentamente (depende de cada placa mãe). Deixe sempre o *TURBO* ligado para seu computador trabalhar na velocidade máxima de processamento.

Botão RESET Reinicia o computador. Quando o computador é reiniciado, uma nova partida é feita (é como se nós ligássemos novamente o computador). Este botão é um dos mais usados por usuários `Windows` dentre os botões localizados no painel do microcomputador. No `GNU/Linux` é raramente usado (com menos freqüência que a tecla `SCROLL LOCK`). É recomendado se pressionar as teclas <CTRL> <ALT> <DEL> para reiniciar o computador e o botão *RESET* somente em último caso, pois o <CTRL> <ALT> <DEL> avisa ao Linux que o usuário pediu para o sistema ser reiniciado assim ele poderá salvar os arquivos, fechar programas e tomar outras providências antes de resetar o computador.

KEYLOCK Permite `ligar/desligar` o teclado. É acionado por uma chave e somente na posição "Cadeado Aberto" permite a pessoa usar o teclado (usar o computador). Alguns computadores não possuem KEYLOCK.

LED POWER Led (normalmente verde) no painel do computador que quando aceso, indica que o computador está ligado. O led é um diodo emissor de luz (light emission diode) que emite luz fria.

LED TURBO Led (normalmente amarelo) no painel do computador. Quando esta aceso, indica que a chave turbo está ligada e o computador funcionando a toda velocidade. Raramente as placas mãe Pentium e acima usam a chave turbo. Mesmo que exista no gabinete do micro, encontra-se desligada.

LED HDD Led (normalmente vermelho) no painel do computador. Acende quando o disco rígido (ou discos) do computador esta sendo usado. Também acende quando uma unidade de CD-ROM está conectada na placa mãe e for usado.

1.9.3 Monitor de Vídeo

O monitor de vídeo se divide em dois tipos:

- *Monocromático* - Mostra tons de cinza
- *Policromático* - A conhecida tela colorida

Quanto ao padrão do monitor, existem diversos:

***CGA* - Color Graphics Adapter** Capacidade de mostrar 4 cores simultâneas em modo gráfico. Uma das primeiras usadas em computadores PCs, com baixa qualidade de imagem, poucos programas funcionavam em telas CGA, quase todos em modo texto. Ficou muito conhecida como "tela verde" embora existem modelos CGA preto e branco.

Hércules Semelhante ao CGA. Pode mostrar 2 cores simultâneas em modo gráfico. A diferença é que apresenta uma melhor qualidade para a exibição de gráficos mas por outro lado, uma grande variedade de programas para monitores CGA não funcionam com monitores Hércules por causa de seu modo de vídeo. Também é conhecido por sua imagem `amarela`. Dependendo da placa de vídeo, você pode configurar um monitor Hércules monocromático para trabalhar como *CGA*.

***EGA* - Enhanced Graphics Adapter** Capacidade de mostrar 16 cores simultâneas em modo gráfico. Razoável melhora da qualidade gráfica, mais programas rodavam neste tipo de tela. Ficou mais conhecida após o lançamento dos computadores 286, mas no Brasil ficou pouco conhecida pois logo em seguida foi lançada o padrão VGA.

***VGA* - Video Graphics Array** Capacidade de mostrar 256 cores simultâneas. Boa qualidade gráfica, este modelo se mostrava capaz de rodar tanto programas texto como gráficos com ótima qualidade de imagem. Se tornou o padrão mínimo para rodar programas em

modo gráfico.

***SVGA* - Super Video Graphics Array** Atual padrão de mercado, capaz de mostrar até 16 milhões de cores simultâneas. Excelente qualidade gráfica, também capaz de operar corretamente em modo texto.

1.10 Placa Mãe

É a placa principal do sistema onde estão localizados o Processador, Memória RAM, Memória Cache, BIOS, CMOS, RTC, etc. A placa mãe possui encaixes onde são inseridas placas de extensão (para aumentar as funções do computador). Estes encaixes são chamados de "`SLOTS`".

1.10.1 Alguns componentes da placa mãe

Abaixo a descrição de alguns tipos de componentes eletrônicos que estão presentes na placa mãe. Não se preocupe se não entender o que eles significam agora:

- `RAM` - Memória de Acesso Aleatório (Randomic Access Memory). É uma memória de armazenamento temporário dos programas e depende de uma fonte de energia para o armazenamento dos programas. É uma memória eletrônica muito rápida assim os programas de computador são executados nesta memória. Seu tamanho é medido em Kilobytes, Megabytes ou Gigabytes.

  Os chips de memória *RAM* podem ser independentes (usando circuitos integrados encaixados em soquetes na placa mãe) ou agrupados placas de 30 pinos, 72 pinos e 168 pinos.

  Quanto maior o tamanho da memória, mais espaço o programa terá ao ser executado. O tamanho de memória RAM pedido por cada programa varia, o `GNU/Linux` precisa de no mínimo 8 MB de memória RAM para ser executado pelo processador.

- `PROCESSADOR` - É a parte do computador responsável pelo processamentos das instruções matemáticas/lógicas e programas carregados na memória *RAM*.

- `CO-PROCESSADOR` - Ajuda o Processador principal a processar as instruções matemáticas. É normalmente embutido no Processador principal em computadores a partir do *486 DX2-66*. Em processadores Pentium e superiores, o co-processador é sempre embutido no processador.

- `CACHE` - Memória de Armazenamento Auxiliar do Processador. Possui alta velocidade de funcionamento, normalmente a mesma que o processador. Serve para aumentar o desempenho de processamento. A memória Cache pode ser embutida na placa mãe ou encaixada externamente através de módulos L2.

- `BIOS` - É a memória *ROM* que contém as instruções básicas para a inicialização do computador, reconhecimento e ativação dos periféricos conectados a placa mãe. As *BIOS*

mais modernas (a partir do 286) também trazem um programa que é usado para configurar o computador modificando os valores localizados na *CMOS*.

As placas controladoras SCSI possuem sua própria *BIOS* que identificam automaticamente os periféricos conectados a ela. Os seguintes tipos de chips podem ser usados para gravar a *BIOS*:

  – `ROM` - Memória Somente para Leitura (Read Only Memory). Somente pode ser lida. É programada de fábrica através de programação elétrica ou química.
  – `PROM` - Memória Somente para Leitura Programável (Programable Read Only Memory) idêntica a *ROM* mas que pode ser programada apenas uma vez por máquinas "Programadoras PROM". É também chamada de `MASK ROM`.
  – `EPROM` - Memória semelhante a *PROM*, mas seu conteúdo pode ser apagado através raios ultra-violeta.
  – `EEPROM` - Memória semelhante a *PROM*, mas seu conteúdo pode ser apagado e regravado. Também é chamada de *Flash*.

- `CMOS` - É uma memória temporária alimentada por uma Bateria onde são lidas/armazenadas as configurações do computador feitas pelo programa residente na BIOS.

1.11 Memória do Computador

A memória é a parte do computador que permitem o armazenamento de dados. A memória é dividida em dois tipos: Principal e Auxiliar. Normalmente quando alguém fala em "memória de computador" está se referindo a memória "Principal". Veja abaixo as descrições de *Memória Principal* e *Auxiliar*.

1.11.1 Memória Principal

É um tipo de memória eletrônica que depende de uma fonte de energia para manter os dados armazenados e perde os dados quando a fonte de energia é desligada. A memória *RAM* do computador (Randomic Access Memory - Memória de Acesso aleatório) é o principal exemplo de memória de armazenamento Principal.

Os dados são armazenados em circuitos integrados ("chips") e enquanto você está usando seu computador, a *RAM* armazena e executa seus programas. Os programas são executados na memória *RAM* porque a memória eletrônica é muito rápida. As memórias EDO, DIMM, DDR, DDR2, DDR3 são exemplos de memória *RAM*.

Se desligarmos o computador ou ocorrer uma queda de energia, você perderá os programas que estiverem em execução ou o trabalho que estiver fazendo. Por esse motivo é necessário o uso de uma memória auxiliar (veja 'Memória Auxiliar' on the next page).

1.11.2 Memória Auxiliar

São dispositivos que NÃO dependem de uma fonte de energia para manter os dados armazenados, os dados não são perdidos quando a fonte de energia é desligada. As *Memórias Auxiliares* são muito mais lentas que as *Memórias Principais* porque utilizam mecanismos mecânicos e elétricos (motores e eletroímãs) para funcionar e fazer a leitura/gravação dos dados. Existem também modelos chamados disco de estado sólido (SSD), os dados são armazenados em chips eletrônicos ao invés de mecanismos mecânicos.

Um exemplo de dispositivos de armazenamento auxiliar são os pen drives, disquetes, cartões SD, discos rígidos, unidades de fita, Zip Drives, DVD/CD/BluRay, etc.

A *Memória Auxiliar* resolve o problema da perda de dados causado pela *Memória Principal* quando o computador é desligado, desta forma podemos ler nossos arquivos e programas da *memória Auxiliar* e copia-los para a *Memória Principal* (memória RAM) para que possam ser novamente usados.

Um exemplo simples é de quando estiver editando um texto e precisar salva-lo, o que você faz é simplesmente salvar os dados da memória *RAM* que estão sendo editados para o disco rígido, desta forma você estará guardando seu documento na *Memória Auxiliar*.

Este tipo de memória é mais lento que a memória principal, é por este motivo que os programas somente são carregados e executados na *Memória Principal*.

1.12 Discos

Os discos são memórias de armazenamento Auxiliares. Entre os vários tipos de discos existentes, posso citar os Flexíveis, Rígidos, Pen-drives, SSD e CDs. Veja as explicações sobre cada um deles abaixo.

1.12.1 Discos Flexíveis

São discos usados para armazenar e transportar pequenas quantidades de dados. Este tipo de disco é normalmente encontrado no tamanho 3 1/2 (1.44MB) polegadas e 5 1/4 polegadas (360Kb ou 1.2MB). Hoje os discos de 3 1/2 são os mais utilizados por terem uma melhor proteção por causa de sua capa plástica rígida, maior capacidade e o menor tamanho o que facilita seu transporte.

Os disquetes são inseridos em um compartimento chamado de "`Unidade de Disquetes`" ou "`Drive`" que faz a leitura/gravação do disquete.

Sua característica é a baixa capacidade de armazenamento e baixa velocidade no acesso aos dados mas podem ser usados para transportar os dados de um computador a outro com grande facilidade. Os disquetes de computador comuns são discos flexíveis.

1.12.2 Disco Rígido

É um disco localizado dentro do computador. É fabricado com discos de metal recompostos por material magnético onde os dados são gravados através de cabeças e revestido externamente por uma proteção metálica que é preso ao gabinete do computador por parafusos. Também é chamado de HD (Hard Disk) ou Winchester. É nele que normalmente gravamos e executamos nossos programas mais usados.

Existe também um tipo de disco rígido chamado *SSD* (disco de estado sólido). A diferença deste disco para o disco rígido comum, é que no SSD os dados são armazenados em chips ao invés de disco magnético.

A característica deste tipo de disco é a alta capacidade de armazenamento de dados e alta velocidade no acesso aos dados.

1.12.3 CD/DVD/BluRay

É um tipo de disco que permite o armazenamento de dados através de um *compact disc* e os dados são lidos através de uma lente ótica. A Unidade de CD é localizada no gabinete do computador e pode ler CDs de músicas, arquivos, interativos, etc. Existem diversos tipos de CDs no mercado, entre eles:

- `CD-R` - CD gravável, pode ser gravado apenas uma vez. Possui sua capacidade de armazenamento entre `600MB` e `740MB` dependendo do formato de gravação usado. Usa um formato lido por todas as unidades de CD-ROM disponíveis no mercado.
- `CD-RW` - CD regravável, pode ser gravado várias vezes, ter seus arquivos apagados, etc. Seu uso é semelhante ao de um disquete de alta capacidade. Possui capacidade de armazenamento de normalmente `640MB` mas isto depende do fabricante. Usa um formato que é lido apenas por unidades leitoras e gravadoras multiseção.
- `DVD-ROM` - Alta capacidade de armazenamento. Pode armazenar até 8GB de arquivos ou programas quando usado em dual layer. `BluRay` - Alta capacidade de armazenamento. Pode armazenar mais de 50GB de arquivos ou programas quando usado em dual layer. É um tipo de CD muito novo no mercado e ainda em desenvolvimento. É lido somente por unidades próprias para este tipo de disco.

1.13 Cuidados Básicos com o Computador

Abaixo uma lista de cuidados básicos para garantir uma melhor conservação e funcionamento de seu computador.

- Não deixe seu computador em locais expostos a umidade ou sol. O mesmo se aplica a mídias como pen-drives, gavetas de HD, cartões de memória etc.
- Limpe o Gabinete e o Monitor com um pano levemente umedecido em água com sabão neutro ou solução de limpeza apropriada para micros. Não use Álcool, querosene, acetona ou qualquer outro tipo de produto abrasivo. O uso de um destes podem estragar o

gabinete de seu computador e se um destes produtos atingir a parte interna pode causar problemas nas placas ou até um incêndio!

- Não retire o Pino central da tomada do computador, ele não veio sobrando e tem utilidade! Este pino é ligado a carcaça do computador (chassis) e deve ser ligado ao terra de sua rede elétrica. As descargas elétricas vindas da fonte e componentes do micro são feitas no chassis e se este pino for retirado você poderá tomar choques ao tocar em alguma parte metálica do micro e queimar componentes sensíveis como o disco rígido, placa mãe, etc.

  Se estiver em dúvida consulte um eletricista de confiança.

- Não instale seu computador muito próximo de campos magnéticos com televisores, aparelhos de som, motores, etc. Estes aparelhos geram ruídos elétricos e/ou magnéticos que podem prejudicar o bom funcionamento de seu micro. OBS: As caixas de som de kits multimídia possuem os ímãs revestidos de metais em seus auto-falantes para não causar nenhuma interferência ao computador.

- Não use a bandeja da unidade de CD/DVD como porta copos!

- Não coloque objetos dentro da unidade de disquetes.

- Antes de desligar seu computador, utilize o comando "`shutdown -h now`" (ou seus sinonimos, como "`halt`", `poweroff`) para desligar corretamente o computador. Este comando finaliza adequadamente os programas, salva os dados, desmontar os sistemas de arquivos `GNU/Linux`. Para detalhes veja 'Desligando o computador' on the following page.

1.14 Dispositivos de Entrada e Saída

- `Entrada` - Permite a comunicação do usuário com o computador. São dispositivos que `enviam` dados ao computador para processamento. Exemplos: Teclado, mouse, touch screen, caneta ótica, scanner.

  O dispositivo de entrada padrão (stdin) em sistemas `GNU/Linux` é o teclado.

- `Saída` - Permite a comunicação do computador com o usuário. São dispositivos que permitem o usuário visualizar o resultado do processamento enviado ao computador. Exemplos: Monitor, Impressora, Plotter, som.

  O dispositivo de saída padrão (stdout) em sistemas `GNU/Linux` é o Monitor.

1.15 Ligando o computador

Para ligar o computador pressione o botão *POWER* ou *I/O* localizado em seu painel frontal do micro.

Imediatamente entrará em funcionamento um programa residente na memória *ROM* (Read Only Memory - memória somente para leitura) da placa mãe que fará os testes iniciais para verificar se os principais dispositivos estão funcionando em seu computador (memória RAM, discos, processador, portas de impressora, memória cache, etc).

Quando o ROM termina os testes básicos, ele inicia a procura do setor de boot nos discos do computador que será carregado na memória RAM do computador. Após carregar o setor de boot, o sistema operacional será iniciado (veja 'Sistema Operacional' on page 4). O setor de boot contém a porção principal usada para iniciar o sistema operacional.

No `GNU/Linux`, o setor de boot normalmente é criado por um gerenciador de inicialização (um programa que permite escolher qual sistema operacional será iniciado). Deste modo podemos usar mais de um sistema operacional no mesmo computador (como o DOS e Linux). Os gerenciadores de inicialização mais usados em sistemas `GNU/Linux` na plataforma Intel X86 são o `GRUB` e o `LILO`.

Caso o ROM não encontre o sistema operacional em nenhum dos discos, ele pedirá que seja inserido um disquete contendo o Sistema Operacional para partida.

1.16 Desligando o computador

Para desligar o computador primeiro digite (como root): "`shutdown -h now`", "`halt`" ou "`poweroff`", o `GNU/Linux` finalizará os programas e gravará os dados em seu disco rígido, quando for mostrada a mensagem "`power down`", pressione o botão *POWER* em seu gabinete para desligar a alimentação de energia do computador.

NUNCA desligue diretamente o computador sem usar o comando `shutdown`, `halt` ou `poweroff`, pois podem ocorrer perda de dados ou falhas no sistema de arquivos de seu disco rígido devido a programas abertos e dados ainda não gravados no disco.

Salve seus trabalhos para não correr o risco de perde-los durante o desligamento do computador.

1.17 Reiniciando o computador

Reiniciar quer dizer iniciar novamente o sistema. Não é recomendável desligar e ligar constantemente o computador pelo botão `ON/OFF`, por este motivo existe recursos para reiniciar o sistema sem desligar o computador. No `GNU/Linux` você pode usar o comando `reboot`, `shutdown -r now` e também pressionar simultaneamente as teclas <CTRL> <ALT> <DEL> para reiniciar de uma forma segura.

Observações:

- Salve seus trabalhos para não correr o risco de perde-los durante a reinicialização do sistema.
- O botão reset do painel frontal do computador também reinicia o computador, mas de uma maneira mais forte pois está ligado diretamente aos circuitos da placa mãe e o

sistema será reiniciado imediatamente, não tendo nenhuma chance de finalizar corretamente os programas, gravar os dados da memória no disco e desmontar os sistemas de arquivos. O uso indevido da tecla reset pode causar corrompimentos em seus arquivos e perdas. Prefira o método de reinicialização explicado acima e use o botão reset somente em último caso.

Capítulo 2

Explicações Básicas

Este capítulo traz explicações sobre os principais componentes existentes no computador e do sistema operacional `Linux`.

2.1 Hardware e Software

Hardware - Significa parte física do computador (disquete, pen-drive, impressoras, monitores, placa mãe, placa de fax, discos rígidos, etc).

Software - São os programas usados no computador (sistema operacional, processador de textos, planilha, banco de dados, scripts, comandos, etc).

2.2 Arquivos

É onde gravamos nossos dados. Um arquivo pode conter um texto feito por nós, uma música, programa, planilha, etc.

Cada arquivo deve ser identificado por um `nome`, assim ele pode ser encontrado facilmente quando desejar usa-lo. Se estiver fazendo um trabalho de história, nada melhor que salva-lo com o nome `historia`. Um arquivo pode ser binário ou texto (para detalhes veja 'Arquivo texto e binário' on page 25).

O `GNU/Linux` é *Case Sensitive* ou seja, ele diferencia letras *maiúsculas* e *minúsculas* nos arquivos. O arquivo `historia` é completamente diferente de `Historia`. Esta regra também é válido para os *comandos* e *diretórios*. Prefira, sempre que possível, usar letras minúsculas para identificar seus arquivos, pois quase todos os comandos do sistema estão em *minúsculas*.

Um arquivo oculto no `GNU/Linux` é identificado por um "." no inicio do nome (por exemplo, `.bashrc`). Arquivos ocultos não aparecem em listagens normais de diretórios, deve ser usado o comando `ls -a` para também listar arquivos ocultos.

2.2.1 Extensão de arquivos

A extensão serve para identificar o tipo do arquivo. A extensão são as letras após um "." no nome de um arquivo, explicando melhor:

- `relatório`**`.txt`** - O `.txt` indica que o conteúdo é um arquivo texto.
- `script`**`.sh`** - Arquivo de Script (interpretado por `/bin/sh`).
- `system`**`.log`** - Registro de algum programa no sistema.
- `arquivo`**`.gz`** - Arquivo compactado pelo utilitário `gzip`.
- `index`**`.html`** - Página de Internet (formato Hypertexto).

A extensão de um arquivo também ajuda a saber o que precisamos fazer para abri-lo. Por exemplo, o arquivo `relatório.txt` é um texto simples e podemos ver seu conteúdo através do comando 'cat' on page 65, já o arquivo `index.html` contém uma página de Internet e precisaremos de um navegador para poder visualiza-lo (como o `lynx`, `Firefox` ou o `Konqueror`).

A extensão (na maioria dos casos) não é requerida pelo sistema operacional `GNU/Linux`, mas é conveniente o seu uso para determinarmos facilmente o tipo de arquivo e que programa precisaremos usar para abri-lo.

2.2.2 Tamanho de arquivos

A unidade de medida padrão nos computadores é o `bit`. A um conjunto de 8 bits nós chamamos de `byte`. Cada arquivo/diretório possui um tamanho, que indica o espaço que ele ocupa no disco e isto é medido em `bytes`. O byte representa uma letra. Assim, se você criar um arquivo vazio e escrever o nome `Linux` e salvar o arquivo, este terá o tamanho de 5 bytes. Espaços em branco e novas linhas também ocupam bytes.

Além do byte existem as medidas Kbytes, Mbytes, Gbytes. Os prefixos K (quilo), M (mega), G (giga), T (tera) etc. vêem da matemática. O "K" significa multiplicar por 10^3, o "M" por 10^6, e assim por diante. Esta letras servem para facilitar a leitura em arquivos de grande tamanho. Um arquivo de 1K é a mesma coisa de um arquivo de 1024 bytes. Uma forma que pode inicialmente lhe ajudar a lembrar: K vem de Kilo que é igual a 1000 - 1Kilo é igual a 1000 gramas certo?.

Da mesma forma 1Mb (ou 1M) é igual a um arquivo de 1024K ou 1.048.576 bytes

1Gb (ou 1G) é igual a um arquivo de 1024Mb ou 1048576Kb ou 1.073.741.824 bytes (1 Gb é igual a 1.073.741.824 bytes, são muitos números!). Deu pra notar que é mais fácil escrever e entender como 1Gb do que 1.073.741.824 bytes :-)

A lista completa em ordem progressiva das unidades de medida é a seguinte:

```
Símbolo 10^ 2^ Nome

K   3 10 Quilo
M   6 20 Mega
G   9 30 Giga
T  12 40 Tera
P  15 50 Peta
E  18 60 Eta
Z  21 70 Zetta
Y  24 80 Yotta
```

2.2.3 Arquivo texto e binário

Quanto ao tipo, um arquivo pode ser de texto ou binário:

`texto` Seu conteúdo é compreendido pelas pessoas. Um arquivo texto pode ser uma carta, um script, um programa de computador escrito pelo programador, arquivo de configuração, etc.

`binário` Seu conteúdo somente pode ser entendido por computadores. Contém caracteres incompreensíveis para pessoas normais. Um arquivo binário é gerado através de um arquivo de programa (digitado pela pessoa que o criou, o programador) através de um processo chamado de `compilação`. Compilação é basicamente a conversão de um programa em linguagem humana para a linguagem de máquina.

2.3 Diretório

Diretório é o local utilizado para armazenar conjuntos arquivos para melhor organização e localização. O diretório, como o arquivo, também é "*Case Sensitive*" (diretório `/teste` é completamente diferente do diretório `/Teste`).

Não podem existir dois arquivos com o `mesmo` nome em um diretório, ou um sub-diretório com um mesmo nome de um arquivo em um mesmo diretório.

Um diretório nos sistemas `Linux/UNIX` são especificados por uma "/" e não uma "\" como é feito no `DOS`. Para detalhes sobre como criar um diretório, veja o comando `mkdir` ('mkdir' on page 63).

2.3.1 Diretório Raíz

Este é o diretório principal do sistema. Dentro dele estão todos os diretórios do sistema. O diretório Raíz é representado por uma "/", assim se você digitar o comando `cd /` você estará acessando este diretório.

Nele estão localizados outros diretórios como o /bin, /sbin, /usr, /usr/local, /mnt, /tmp, /var, /home, etc. Estes são chamados de *sub-diretórios* pois estão dentro do diretório "/". A estrutura de *diretórios* e *sub-diretórios* pode ser identificada da seguinte maneira:

- /
- /bin
- /sbin
- /usr
- /usr/local
- /mnt
- /tmp
- /var
- /home

A estrutura de diretórios também é chamada de Árvore de Diretórios porque é parecida com uma *árvore* de cabeça para baixo. Cada diretório do sistema tem seus respectivos arquivos que são armazenados conforme regras definidas pela *FHS* (*FileSystem Hierarchy Standard - Hierarquia Padrão do Sistema de Arquivos*) versão 2.0, definindo que tipo de arquivo deve ser armazenado em cada diretório.

2.3.2 Diretório atual

É o diretório em que nos encontramos no momento. Você pode digitar pwd (veja 'pwd' on page 63) para verificar qual é seu diretório atual.

O diretório atual também é identificado por um "." (ponto). O comando comando `ls .` pode ser usado para listar seus arquivos (é claro que isto é desnecessário porque se não digitar nenhum diretório, o comando ls listará o conteúdo do diretório atual).

2.3.3 Diretório home

Também chamado de diretório de usuário. Em sistemas GNU/Linux cada usuário (inclusive o root) possui seu próprio diretório onde poderá armazenar seus programas e arquivos pessoais.

Este diretório está localizado em /home/[login], neste caso se o seu login for "joao" o seu diretório home será /home/joao. O diretório home também é identificado por um ~(til), você pode digitar tanto o comando `ls /home/joao` como `ls ~` para listar os arquivos de seu diretório home.

O diretório home do usuário root (na maioria das distribuições GNU/Linux) está localizado em /root.

Dependendo de sua configuração e do número de usuários em seu sistema, o diretório de usuário pode ter a seguinte forma: /home/[1letra_do_nome]/[login], neste caso se o seu login for "joao" o seu diretório home será /home/j/joao.

2.3.4 Diretório Superior

O diretório superior (Upper Directory) é identificado por `..` (2 pontos).

Caso estiver no diretório `/usr/local` e quiser listar os arquivos do diretório `/usr` você pode digitar, `ls ..` Este recurso também pode ser usado para copiar, mover arquivos/diretórios, etc.

2.3.5 Diretório Anterior

O diretório anterior é identificado por "-". É útil para retornar ao último diretório usado.

Se estive no diretório `/usr/local` e digitar `cd /lib`, você pode retornar facilmente para o diretório `/usr/local` usando `cd -`.

2.3.6 Caminho na estrutura de diretórios

São os diretórios que teremos que percorrer até chegar no arquivo ou diretório que que procuramos. Se desejar ver o arquivo `/etc/hosts` você tem duas opções:

1 Mudar o diretório padrão para `/etc` com o comando `cd /etc` e usar o comando `cat hosts`

2 Usar o comando "`cat`" especificando o caminho completo na estrutura de diretórios e o nome de arquivo: `cat /etc/hosts`.

As duas soluções acima permitem que você veja o arquivo `GPL`. A diferença entre as duas é a seguinte:

- Na primeira, você muda o diretório padrão para `/usr/doc/copyright` (confira digitando `pwd`) e depois o comando `cat GPL`. Você pode ver os arquivos de `/usr/doc/copyright` com o comando "`ls`". `/usr/doc/copyright` é o caminho de diretório que devemos percorrer para chegar até o arquivo `GPL`.
- Na segunda, é digitado o caminho completo para o "`cat`" localizar o arquivo `GPL`: `cat /usr/doc/copyright/GPL`. Neste caso, você continuará no diretório padrão (confira digitando `pwd`). Digitando `ls`, os arquivos do diretório atual serão listados.

O *caminho de diretórios* é necessário para dizer ao sistema operacional onde encontrar um arquivo na "árvore" de diretórios.

2.3.7 Exemplo de diretório

Um exemplo de diretório é o seu diretório de usuário, todos seus arquivos essenciais devem ser colocadas neste diretório. Um diretório pode conter outro diretório, isto é útil quando temos muitos arquivos e queremos melhorar sua organização. Abaixo um exemplo de uma empresa que precisa controlar os arquivos de Pedidos que emite para as fábricas:

/pub/vendas - diretório principal de vendas /pub/vendas/mes01-1999 - diretório contendo vendas do mês 01/1999 /pub/vendas/mes02-2009 - diretório contendo vendas do mês 02/2009 /pub/vendas/mes01-2010 - diretório contendo vendas do mês 03/2010

mes01-99,mes02-2009,mes01-2010 são diretórios usados para armazenar os arquivos de pedidos do mês e ano correspondente. Isto é essencial para organização, pois se todos os pedidos fossem colocados diretamente no diretório vendas, seria muito difícil encontrar o arquivo do cliente "João" do mês 01/2009.

Você deve ter reparado que usei a palavra *sub-diretório* para mes01-1999, mes02-2009 e mes01-2010, porque que eles estão dentro do diretório vendas. Da mesma forma, vendas é um sub-diretório de pub.

2.3.8 Estrutura básica de diretórios do Sistema Linux

O sistema GNU/Linux possui a seguinte estrutura básica de diretórios organizados segundo o FHS (Filesystem Hierarchy Standard):

/bin Contém arquivos programas do sistema que são usados com freqüência pelos usuários.

/boot Contém arquivos necessários para a inicialização do sistema.

/cdrom Ponto de montagem da unidade de CD-ROM.

/media Ponto de montagem de dispositivos diversos do sistema (rede, pen-drives, CD-ROM em distribuições mais novas).

/dev Contém arquivos usados para acessar dispositivos (periféricos) existentes no computador.

/etc Arquivos de configuração de seu computador local.

/floppy Ponto de montagem de unidade de disquetes

/home Diretórios contendo os arquivos dos usuários.

/lib Bibliotecas compartilhadas pelos programas do sistema e módulos do kernel.

/lost+found Local para a gravação de arquivos/diretórios recuperados pelo utilitário fsck.ext2. Cada partição possui seu próprio diretório lost+found.

/mnt Ponto de montagem temporário.

/proc Sistema de arquivos do kernel. Este diretório não existe em seu disco rígido, ele é colocado lá pelo kernel e usado por diversos programas que fazem sua leitura, verificam configurações do sistema ou modificar o funcionamento de dispositivos do sistema através da alteração em seus arquivos.

/sys Sistema de arquivos do kernel. Este diretório não existe em seu disco rígido, ele é colocado lá pelo kernel e usado por diversos programas que fazem sua leitura, verificam configurações do sistema ou modificar o funcionamento de dispositivos do sistema através da alteração em seus arquivos.

/root Diretório do usuário root.

/sbin Diretório de programas usados pelo superusuário (root) para administração e controle do funcionamento do sistema.

/tmp Diretório para armazenamento de arquivos temporários criados por programas.

/usr Contém maior parte de seus programas. Normalmente acessível somente como leitura.

/var Contém maior parte dos arquivos que são gravados com freqüência pelos programas do sistema, e-mails, spool de impressora, cache, etc.

2.4 Nomeando Arquivos e Diretórios

No `GNU/Linux`, os arquivos e diretórios pode ter o tamanho de até *255* letras. Você pode identifica-lo com uma extensão (um conjunto de letras separadas do nome do arquivo por um ".").

Os programas executáveis do `GNU/Linux`, ao contrário dos programas de `DOS` e `Windows`, não são executados a partir de extensões `.exe`, `.com` ou `.bat`. O `GNU/Linux` (como todos os sistemas POSIX) usa a *permissão de execução* de arquivo para identificar se um arquivo pode ou não ser executado.

No exemplo anterior, nosso trabalho de história pode ser identificado mais facilmente caso fosse gravado com o nome `trabalho.text` ou `trabalho.txt`. Também é permitido gravar o arquivo com o nome `TrabalhodeHistoria.txt` mas não é recomendado gravar nomes de arquivos e diretórios com espaços. Porque será necessário colocar o nome do arquivo entre "aspas" para acessa-lo (por exemplo, `cat "Trabalho de Historia.txt"`). Ao invés de usar espaços, prefira *capitalizar* o arquivo (usar letras maiúsculas e minúsculas para identifica-lo): `TrabalhodeHistoria.txt`.

2.5 Comandos

Comandos são ordens que passamos ao sistema operacional para executar uma determinada tarefa.

Cada comando tem uma função específica, devemos saber a função de cada comando e escolher o mais adequado para fazer o que desejamos, por exemplo:

- `ls` - Mostra arquivos de diretórios
- `cd` - Para mudar de diretório

Este guia tem uma lista de vários comandos organizados por categoria com a explicação sobre o seu funcionamento e as opções aceitas (incluindo alguns exemplos).

É sempre usado um espaço depois do comando para separá-lo de uma opção ou parâmetro que será passado para o processamento. Um comando pode receber opções e parâmetros:

opções As *opções* são usadas para controlar como o comando será executado, por exemplo, para fazer uma listagem mostrando o *dono, grupo, tamanho dos arquivos* você deve digitar `ls -l`. Opções podem ser passadas ao comando através de um "-" ou "–":

- **-** Opção identificada por uma letra. Podem ser usadas mais de uma opção com um único hífen. O comando `ls -l -a` é a mesma coisa de `ls -la`
- **–** Opção identificada por um nome. Também chamado de opção extensa. O comando `ls --all` é equivalente a `ls -a`.

Pode ser usado tanto "-" como "–", mas há casos em que somente "-" ou "–" esta disponível.

parâmetros Um parâmetro identifica o *caminho, origem, destino, entrada padrão* ou *saída padrão* que será passada ao comando. Se você digitar: `ls /usr/share/doc/copyright`, `/usr/share/doc/copyright` será o parâmetro passado ao comando `ls`, neste caso queremos que ele liste os arquivos do diretório */usr/share/doc/copyright*. É normal errar o nome de comandos, mas não se preocupe, quando isto acontecer o sistema mostrará a mensagem `command not found` (comando não encontrado) e voltará ao aviso de comando. As mensagens de erro não fazem nenhum mal ao seu sistema, somente dizem que algo deu errado para que você possa corrigir e entender o que aconteceu. No `GNU/Linux`, você tem a possibilidade de criar comandos personalizados usando outros comandos mais simples (isto será visto mais adiante). Os comandos se encaixam em duas categorias: *Comandos Internos* e *Comandos Externos*.

Por exemplo: "`ls -la /usr/share/doc`", `ls` é o comando, `-la` é a opção passada ao comando, e `/usr/share/doc` é o diretório passado como parâmetro ao comando `ls`.

2.5.1 Comandos Internos

São comandos que estão localizados dentro do interpretador de comandos (normalmente o `Bash`) e não no disco. Eles são carregados na memória RAM do computador junto com o interpretador de comandos.

Quando executa um comando, o interpretador de comandos verifica primeiro se ele é um *Comando Interno* caso não seja é verificado se é um *Comando Externo*.

Exemplos de comandos internos são: `cd, exit, echo, bg, fg, source, help`

2.6 Comandos Externos

São comandos que estão localizados no disco. Os comandos são procurados no disco usando o ordem do `PATH` e executados assim que encontrados.

Para detalhes veja 'path' on page 51.

2.7 Aviso de comando (Prompt)

Aviso de comando (ou Prompt), é a linha mostrada na tela para *digitação de comandos* que serão passados ao `interpretador de comandos` para sua execução.

A posição onde o comando será digitado é marcado um "traço" piscante na tela chamado de *cursor*. Tanto em shells texto como em gráficos é necessário o uso do cursor para sabermos onde iniciar a digitação de textos e nos orientarmos quanto a posição na tela.

O aviso de comando do usuário `root` é identificado por uma "#" (tralha), e o aviso de comando de usuários é identificado pelo símbolo "$". Isto é padrão em sistemas `UNIX`.

Você pode retornar comandos já digitados pressionando as teclas `Seta para cima` / `Seta para baixo`.

A tela pode ser rolada para baixo ou para cima segurando a tecla `SHIFT` e pressionando `PGUP` ou `PGDOWN`. Isto é útil para ver textos que rolaram rapidamente para cima.

Abaixo algumas dicas sobre a edição da linha de comandos (não é necessário se preocupar em decora-los):

- Pressione a tecla `Back Space` ("**<–**") para apagar um caracter à esquerda do cursor.
- Pressione a tecla `Del` para apagar o caracter acima do cursor.
- Pressione `CTRL+A` para mover o cursor para o inicio da linha de comandos.
- Pressione `CTRL+E` para mover o cursor para o fim da linha de comandos.
- Pressione `CTRL+U` para apagar o que estiver à esquerda do cursor. O conteúdo apagado é copiado para uso com `CTRL+y`.
- Pressione `CTRL+K` para apagar o que estiver à direita do cursor. O conteúdo apagado é copiado para uso com `CTRL+y`.
- Pressione `CTRL+L` para limpar a tela e manter o texto que estiver sendo digitado na linha de comando (parecido com o comando `clear`).
- Pressione `CTRL+Y` para colocar o texto que foi apagado na posição atual do cursor.

2.8 Interpretador de comandos

Também conhecido como "shell". É o programa responsável em interpretar as instruções enviadas pelo usuário e seus programas ao sistema operacional (o kernel). Ele que executa comandos lidos do dispositivo de entrada padrão (teclado) ou de um arquivo executável. É a principal ligação entre o usuário, os programas e o kernel. O `GNU/Linux` possui diversos tipos de interpretadores de comandos, entre eles posso destacar o `bash, ash, csh, tcsh, sh,` etc. Entre eles o mais usado é o `bash`. O interpretador de comandos do DOS, por exemplo, é o `command.com`.

Os comandos podem ser enviados de duas maneiras para o interpretador: `interativa` e `não-interativa`:

Interativa Os comandos são digitados no aviso de comando e passados ao interpretador de comandos um a um. Neste modo, o computador depende do usuário para executar uma tarefa, ou próximo comando.

Não-interativa São usados arquivos de comandos criados pelo usuário (scripts) para o computador executar os comandos na ordem encontrada no arquivo. Neste modo, o computador executa os comandos do arquivo um por um e dependendo do término do comando, o script pode checar qual será o próximo comando que será executado e dar continuidade ao processamento.

Este sistema é útil quando temos que digitar por várias vezes seguidas um mesmo comando ou para compilar algum programa complexo.

O shell `Bash` possui ainda outra característica interessante: A completação dos nomes. Isto é feito pressionando-se a tecla `TAB`. Por exemplo, se digitar "ls tes" e pressionar <tab>, o `Bash` localizará todos os arquivos que iniciam com "tes" e completará o restante do nome. Caso a completação de nomes encontre mais do que uma expressão que satisfaça a pesquisa, ou nenhuma, é emitido um beep. Se você apertar novamente a tecla TAB imediatamente depois do beep, o interpretador de comandos irá listar as diversas possibilidades que satisfazem a pesquisa, para que você possa escolher a que lhe interessa. A completação de nomes funciona sem problemas para comandos internos.

Exemplo: `ech` (pressione `TAB`). `ls /vm`(pressione `TAB`)

2.9 Terminal Virtual (console)

Terminal (ou console) é o teclado e tela conectados em seu computador. O `GNU/Linux` faz uso de sua característica *multi-usuária* usando os "terminais virtuais". Um terminal virtual é uma segunda seção de trabalho completamente independente de outras, que pode ser acessada no computador local ou remotamente via `telnet, rsh, rlogin`, etc.

No `GNU/Linux`, em modo texto, você pode acessar outros terminais virtuais segurando a tecla `ALT` e pressionando `F1 a F6`. Cada tecla de função corresponde a um número de terminal do 1 ao 6 (o sétimo é usado por padrão pelo ambiente gráfico X). O `GNU/Linux` possui mais de 63 terminais virtuais, mas apenas 6 estão disponíveis inicialmente por motivos de economia de memória RAM .

Se estiver usando o modo gráfico, você deve segurar `CTRL+ ALT` enquanto pressiona uma tela de <F1> a <F6>. Para voltar ao modo gráfico, pressione `CTRL+ALT+` <F7>.

Um exemplo prático: Se você estiver usando o sistema no Terminal 1 com o nome "joao" e desejar entrar como "root" para instalar algum programa, segure `ALT` enquanto pressiona <F2> para abrir o segundo terminal virtual e faça o login como "root". Será aberta uma nova seção para o usuário "root" e você poderá retornar a hora que quiser para o primeiro terminal pressionando `ALT`+<F1>.

2.10 Login

Login é a entrada no sistema quando você digita seu *nome* e *senha*. Por enquanto vou manter o seu suspense sobre o que é o *logout*.

2.11 Logout

Logout é a saída do sistema. A saída do sistema é feita pelos comandos `logout`, `exit`, `CTRL+D`, ou quando o sistema é reiniciado ou desligado.

2.12 Curingas

Curingas (ou referência global) é um recurso usado para especificar um ou mais arquivos ou diretórios do sistema de uma só vez. Este é um recurso permite que você faça a filtragem do que será listado, copiado, apagado, etc. São usados 4 tipos de curingas no `GNU/Linux`:

- "*" - Faz referência a um nome completo/restante de um arquivo/diretório.
- "?" - Faz referência a uma letra naquela posição.
- `[padrão]` - Faz referência a uma faixa de caracteres de um arquivo/diretório. Padrão pode ser:
  - `[a-z][0-9]` - Faz referência a caracteres de `a` até `z` seguido de um caracter de `0` até `9`.
  - `[a,z][1,0]` - Faz a referência aos caracteres `a` e `z` seguido de um caracter `1` ou `0` naquela posição.
  - `[a-z,1,0]` - Faz referência a intervalo de caracteres de `a` até `z` ou `1` ou `0` naquela posição.

  A procura de caracteres é "Case Sensitive" assim se você deseja que sejam localizados todos os caracteres alfabéticos você deve usar `[a-zA-Z]`.

  Caso a expressão seja precedida por um `^`, faz referência a qualquer caracter exceto o da expressão. Por exemplo `[^abc]` faz referência a qualquer caracter exceto `a`, `b` e `c`.

- `{padrões}` - Expande e gera strings para pesquisa de padrões de um arquivo/diretório.
  - `X{ab,01}` - Faz referência a seqüencia de caracteres `Xab` ou `X01`
  - `X{a-z,10}` Faz referencia a seqüencia de caracteres X`a-z` e `X10`.

O que diferencia este método de expansão dos demais é que a existência do arquivo/diretório é opcional para geração do resultado. Isto é útil para a criação de diretórios. Lembrando que os 4 tipos de curingas ("*", "?", "[]", "{}") podem ser usados juntos. Para entender melhor vamos a prática:

Vamos dizer que tenha 5 arquivo no diretório `/usr/teste`: `teste1.txt,teste2.txt, teste3.txt,teste4.new,teste5.new`.

Caso deseje listar **todos** os arquivos do diretório `/usr/teste` você pode usar o coringa "*" para especificar todos os arquivos do diretório:

`cd /usr/teste` e `ls *` ou `ls /usr/teste/*`.

Não tem muito sentido usar o comando `ls` com "*" porque todos os arquivos serão listados se o `ls` for usado sem nenhum Coringa.

Agora para listar todos os arquivos `teste1.txt,teste2.txt,teste3.txt` com excessão de `teste4.new`, `teste5.new`, podemos usar inicialmente 3 métodos:

1 Usando o comando `ls *.txt` que pega todos os arquivos que começam com qualquer nome e terminam com `.txt`.

2 Usando o comando `ls teste?.txt`, que pega todos os arquivos que começam com o nome `teste`, tenham qualquer caracter no lugar do coringa `?` e terminem com `.txt`. Com o exemplo acima `teste*.txt` também faria a mesma coisa, mas se também tivéssemos um arquivo chamado `teste10.txt` este também seria listado.

3 Usando o comando `ls teste[1-3].txt`, que pega todos os arquivos que começam com o nome `teste`, tenham qualquer caracter entre o número 1-3 no lugar da 6a letra e terminem com `.txt`. Neste caso se obtém uma filtragem mais exata, pois o coringa *?* especifica qualquer caracter naquela posição e [] especifica números, letras ou intervalo que será usado.

Agora para listar somente `teste4.new` e `teste5.new` podemos usar os seguintes métodos:

1 `ls *.new` que lista todos os arquivos que terminam com `.new`

2 `ls teste?.new` que lista todos os arquivos que começam com `teste`, contenham qualquer caracter na posição do coringa *?* e terminem com `.new`.

3 `ls teste[4,5].*` que lista todos os arquivos que começam com `teste` contenham números de 4 e 5 naquela posição e terminem com qualquer extensão.

Existem muitas outras formas de se fazer a mesma coisa, isto depende do gosto de cada um. O que pretendi fazer aqui foi mostrar como especificar mais de um arquivo de uma só vez. O uso de curingas será útil ao copiar arquivos, apagar, mover, renomear, e nas mais diversas partes do sistema. Alias esta é uma característica do `GNU/Linux`: permitir que a mesma coisa possa ser feita com liberdade de várias maneiras diferentes.

Capítulo 3

Para quem esta migrando (ou pensando em migrar) do DOS/Windows para o Linux

Este capítulo explica as diferenças e particularidades do sistema GNU/Linux comparado ao Windows, DOS e uma lista de equivalência entre comandos e programas executados no CMD do Windows/DOS e GNU/Linux, que pode servir de comparação para que o usuário possa conhecer e utilizar os comandos/programas GNU/Linux que tem a mesma função no ambiente DOS/Windows.

3.1 Quais as diferenças iniciais

- Quando entrar pela primeira vez no GNU/Linux (ou qualquer outro UNIX, a primeira coisa que verá será a palavra login: escrita na tela.

  A sua aventura começa aqui, você deve ser uma pessoa cadastrada no sistema (ter uma conta) para que poder entrar. No login você digita seu nome (por exemplo, gleydson) e pressiona Enter. Agora será lhe pedida a senha, repare que a senha não é mostrada enquanto é digitada, isto serve de segurança e para enganar pessoas que estão próximas de você "tocando" algumas teclas a mais enquanto digita a senha e fazendo-as pensar que você usa uma grande senha ;-) (com os asteriscos aparecendo isto não seria possível).

  Caso cometa erros durante a digitação da senha, basta pressionar a tecla Back Space para apagar o último caracter digitado e terminar a entrada da senha.

  Pressione Enter, se tudo ocorrer bem você estará dentro do sistema e será presenteado com o símbolo # (caso tenha entrado como usuário root) ou $ (caso tenha entrado como um usuário normal).

  Existe um mecanismo de segurança que te alerta sobre eventuais tentativas de entrada no sistema por intrusos usando seu login, faça um teste: entre com seu login e digite a senha errada, na segunda vez entre com a senha correta no sistema. Na penúltima linha

das mensagens aparece uma mensagem "1 failure since last login", o que quer dizer "1 falha desde o último login". Isto significa que alguém tentou entrar 1 vez com seu nome e senha no sistema, sem sucesso.

- A conta `root` não tem restrições de acesso ao sistema e pode fazer tudo o que quiser, é equivalente ao usuário normal do `DOS` e `Windows`. Use a conta `root` somente para manutenções no sistema e instalação de programas, qualquer movimento errado pode comprometer todo o sistema. Para detalhes veja 'A conta root' on page 98.
- No `GNU/Linux` os diretório são identificados por uma / e não por uma \ como acontece no `DOS`. Para entrar no diretório `/bin`, você deve usar `cd /bin`.
- Os comandos são `case-sensitive`, o que significa que ele diferencia as letras maiúsculas de minúsculas em arquivos e diretórios. O comando `ls` e `LS` são completamente diferentes.
- A multitarefa lhe permite usar vários programas simultaneamente (não pense que multitarefa somente funciona em ambientes gráficos, pois isto é errado!). Para detalhes veja 'Execução de programas' on page 51.
- Os dispositivos também são identificados e uma forma diferente que no `DOS` por exemplo:

```
 DOS/Windows        Linux
-------------   ---------------
A:              /dev/fd0
B:              /dev/fd1
C:              /dev/hda1 ou /dev/sda1
LPT1            /dev/lp0
LPT2            /dev/lp1
LPT3            /dev/lp2
COM1            /dev/ttyS0
COM2            /dev/ttyS1
COM3            /dev/ttyS2
COM4            /dev/ttyS3
```

- Os recursos multiusuário lhe permite acessar o sistema de qualquer lugar sem instalar nenhum driver, ou programa gigante, apenas através de conexões TCP/IP, como a Internet. Também é possível acessar o sistema localmente com vários usuários (cada um executando tarefas completamente independente dos outros) através dos Terminais Virtuais. Faça um teste: pressione ao mesmo tempo a tecla `ALT` e `F2` e você será levado para o segundo Terminal Virtual, pressione novamente `ALT` e `F1` para retornar ao anterior.
- Para reiniciar o computador, você pode pressionar CTRL+ALT+DEL (como usuário `root`) ou digitar `shutdown -r now`. Veja 'Reiniciando o computador' on page 20 para detalhes .

- Para desligar o computador, digite `shutdown -h now` e espere o aparecimento da mensagem `Power Down` para apertar o botão LIGA/DESLIGA do computador. Veja 'Desligando o computador' on page 20 para detalhes.

3.2 Comandos equivalentes entre DOS/CMD do Windows e o Linux

Esta seção contém os comandos equivalentes entre estes dois sistemas e a avaliação entre ambos. Grande parte dos comandos podem ser usados da mesma forma que no `DOS`, mas os comandos `Linux` possuem avanços para utilização neste ambiente multiusuário/multitarefa.

O objetivo desta seção é permitir as pessoas com experiência em `DOS` fazer rapidamente no `GNU/Linux` as tarefas que fazem no `DOS`. A primeira coluna tem o nome do comando no `DOS`, a segunda o comando que possui a mesma função no `GNU/Linux` e na terceira coluna as diferenças.

```
  DOS        Linux                           Diferenças
--------   ------------  ---------------------------------------------------
cls        clear         Sem diferenças.
dir        ls -la        A listagem no Linux possui mais campos (as
                         permissões de acesso) e o total de espaço ocupado
                         no diretório e livre no disco deve ser visto
                         separadamente usando o comando du e df.
                         Permite também listar o conteúdo de diversos
                         diretórios com um só comando (ls /bin /sbin /...).
dir/s      ls -lR        Sem diferenças.
dir/od     ls -tr        Sem diferenças.
cd         cd            Poucas diferenças. cd sem parâmetros retorna ao
                         diretório de usuário e também permite o uso
                         de "cd -" para retornar ao diretório anteriormente
                         acessado.
del        rm            Poucas diferenças. O rm do Linux permite
                         especificar diversos arquivos que serão apagados
                         (rm arquivo1 arquivo2 arquivo3). Para ser mostrados
                         os arquivos apagados, deve-se especificar o
                         parâmetro "-v" ao comando, e "-i" para pedir
                         a confirmação ao apagar arquivos.
md         mkdir         Uma só diferença: No Linux permite que vários
                         diretórios sejam criados de uma só vez
                         (mkdir /tmp/a /tmp/b...).
copy       cp            Poucas diferenças. Para ser mostrados os arquivos
                         enquanto estão sendo copiados, deve-se usar a
                         opção "-v", e para que ele pergunte se deseja
                         substituir um arquivo já existente, deve-se usar
                         a opção "-i".
echo       echo          Sem diferenças.
```

```
path      path          No Linux deve ser usado ":" para separar os
                        diretórios e usar o comando
                        "export PATH=caminho1:/caminho2:/caminho3:"
                        para definir a variável de ambiente PATH.
                        O path atual pode ser visualizado através
                        do comando "echo $PATH".
ren       mv            Poucas diferenças. No Linux não é possível
                        renomear vários arquivos de uma só vez
                        (como "ren *.txt *.bak"). É necessário usar
                        um shell script para fazer isto.
type      cat           Sem diferenças.
ver       uname -a      Poucas diferenças (o uname tem algumas opções
                        a mais).
date      date          No Linux mostra/modifica a Data e Hora do sistema.
time      date          No Linux mostra/modifica a Data e Hora do sistema.
attrib    chmod         O chmod possui mais opções por tratar as permissões
                        de acesso de leitura, gravação e execução para
                        donos, grupos e outros usuários.
chkdsk    fsck          O fack é mais rápido e a checagem mais abrangente.
scandisk  fsck          O fsck é mais rápido e a checagem mais abrangente.
doskey    -----         A memorização de comandos é feita automaticamente pelo
                        bash.
edit      vi, ae,       O edit é mais fácil de usar, mas usuário
        emacs, mcedit experientes apreciarão os recursos do vi ou
                        o emacs (programado em lisp).
fdisk    fdisk, cfdisk Os particionadores do Linux trabalham com
                        praticamente todos os tipos de partições de
                        diversos sistemas de arquivos diferentes.
format    mkfs.ext3     Poucas diferenças, precisa apenas que seja
                        especificado o dispositivo a ser formatado
                        como "/dev/fd0" ou "/dev/hda10" (o
                        tipo de identificação usada no Linux), ao
                        invés de "A:" ou "C:".
help      man, info     Sem diferenças.
interlnk  plip          O plip do Linux permite que sejam montadas
                        redes reais a partir de uma conexão via Cabo
                        Paralelo ou Serial. A máquina pode fazer tudo
                        o que poderia fazer conectada em uma rede
                        (na realidade é uma rede e usa o TCP/IP como
                        protocolo) inclusive navegar na Internet, enviar
                        e-mails, irc, etc.
intersvr  plip          Mesmo que o acima.
keyb      loadkeys      Sem diferenças (somente que a posição das
                        teclas do teclado pode ser editada.
                        Desnecessário para a maioria dos usuários).
label     e2label       É necessário especificar a partição que terá
```

```
                          o nome modificado.
mem       cat /proc/meminfo Mostra detalhes sobre a quantidade de dados
          top             em buffers, cache e memória virtual (disco).
more      more, less      O more é equivalente a ambos os sistemas, mas
                          o less permite que sejam usadas as setas para
                          cima e para baixo, o que torna a leitura do
                          texto muito mais agradável.
move      mv              Poucas diferenças. Para ser mostrados os arquivos
                          enquanto estão sendo movidos, deve-se usar a
                          opção "-v", e para que ele pergunte se deseja
                          substituir um arquivo já existente deve-se usar
                          a opção "-i".
scan      clamav          Os principais fabricantes disponibilizam anti-virus
                          para Linux, na maioria das vezes para integrar a
                          servidores de arquivos, e-mails, protegendo estações
                          Windows. Infecções por vírus são raras no Linux devido
                          as restrições do usuário durante execução de
                          programas (quando corretamente utilizadas).
backup    tar             O tar permite o uso de compactação (através do
                          parâmetro -z) e tem um melhor esquema de
                          recuperação de arquivos corrompidos que já
                          segue evoluindo há 30 anos em sistemas UNIX.
print     lpr             O lpr é mais rápido e permite até mesmo
                          impressões de gráficos ou arquivos compactados
                          diretamente caso seja usado o programa
                          magicfilter. É o programa de Spool de
                          impressoras usados no sistema Linux/Unix.
vol       e2label         Sem diferenças.
xcopy     cp -R           Pouca diferença, requer que seja usado a
                          opção "-v" para mostrar os arquivos que
                          estão sendo copiados e "-i" para pedir
                          confirmação de substituição de arquivos.
```

3.2.1 Arquivos de configuração

Os arquivos `config.sys` e `autoexec.bat` são equivalentes aos arquivos do diretório `/etc` especialmente o `/etc/inittab` e arquivos dentro do diretório /etc/init.d .

3.3 Usando a sintaxe de comandos DOS no Linux

Você pode usar os comandos do pacote `mtools` para simular os comandos usados pelo `DOS` no `GNU/Linux`, a diferença básica é que eles terão a letra `m` no inicio do nome. Os seguintes comandos são suportados:

- `mattrib` - Ajusta modifica atributos de arquivos

- mcat - Mostra os dados da unidade de disquete em formato RAW
- mcd - Entra em diretórios
- mcopy - Copia arquivos/diretórios
- mdel - Exclui arquivos
- mdeltree - Exclui arquivos, diretórios e sub-diretórios
- mdir - Lista arquivos e diretórios
- mdu - Mostra o espaço ocupado pelo diretório do DOS
- mformat - Formatador de discos
- minfo - Mostra detalhes sobre a unidade de disquetes
- mlabel - Cria um volume para unidades DOS
- mmd - Cria diretórios
- mmount - Monta discos DOS
- mmove - Move ou renomeia arquivos/subdiretórios
- mpartition - Particiona um disco para ser usado no DOS
- mrd - Remove um diretório
- mren - Renomeia arquivos
- mtype - Visualiza o conteúdo de arquivos (equivalente ao cat)
- mtoolstest - Exibe a configuração atual do mtools
- mshowfat - Mostra a FAT da unidade
- mbadblocks - Procura por setores defeituosos na unidade
- mzip - Altera modo de proteção e ejeta discos em unidades Jaz/ZIP
- mkmanifest - Cria um shell script para restaurar nomes extensos usados no UNIX
- mcheck - Verifica arquivos na unidade

3.4 Programas equivalentes entre Windows/DOS e o Linux

Esta seção contém programas equivalentes para quem está vindo do DOS e Windows e não sabe o que usar no GNU/Linux. Esta seção também tem por objetivo permitir ao usuário que ainda não usa GNU/Linux decidir se a passagem vale a pena vendo se o sistema tem os programas que precisa.

Note que esta listagem mostra os programas equivalentes entre o DOS/Windows e o GNU/Linux cabendo a você a decisão final de migrar ou não. Lembrando que é possível usar o Windows, OS/2, DOS, OS/2 e GNU/Linux no mesmo disco rígido sem qualquer tipo de conflito. A listagem abaixo pode estar incompleta, se encontrar algum programa que não esteja listado aqui, por favor entre em contato pelo E-Mail <gleydson@guiafoca.org> para inclui-lo na listagem.

```
DOS/Windows           Linux                        Diferenças
-----------           ----------         --------------------------------
MS Word               Open Office,       O Open Office possui todos os
                                         recursos do Word além de ter
                                         a interface gráfica igual, menus
                                         e teclas de atalho idênticas ao
                                         Word, o que facilita a migração.
```

		Também trabalha com arquivos no formato Word97/2000 e não é vulnerável a vírus de macro. É distribuído gratuitamente e não requer pagamento de licença podendo ser instalado em quantos computadores você quiser (tanto domésticos como de empresas).
MS Excel	Open Office	Mesmos pontos do acima e também abre arquivos Excel97/2000.
MS PowerPoint	Open Office	Mesmos pontos do acima.
MS Access	MySQL, PostgreSQL Oracle	Existem diversas ferramentas de conceito para bancos de dados corporativos no Linux. Todos produtos compatíveis com outras plataformas.
MS Outlook	Pine, evolution mutt, sylpheed, icedove	Centenas de programas de E-Mail tanto em modo texto como em modo gráfico. Instale, avalie e escolha.
MS Internet Explorer	Firefox, Opera, Mozilla, lynx.	Os três primeiros para modo gráfico e o lynx opera em modo texto.
ICQ	LICQ, PIDGIM, SIM	Muito prático e fácil de operar. Possibilita a mudança completa da aparência do programa através de Skins. A organização dos menus deste programa é outro ponto de destaque.
MSN	AMSN, PIDGIM	Permite conversar diretamente com usuários do Microsoft MSN.
Photo Shop	The Gimp	Fácil de usar, possui muitos scripts que permitem a criação rápida e fácil de qualquer tipo de efeito profissional pelo usuário mais leigo. Acompanha centenas de efeitos especiais e um belo manual em html com muitas fotos (aproximadamente 20MB) que mostra o que é possível se fazer com ele.
Corel Photo Paint	GIMP	Corel Photo-Paint para
Corel Draw	Inkscape, Sodipodi	Programas equivalentes
Autocad	Qcad	Programa com funções genéricas
Visio	dia	Possui funcionalidades identicas

		e ótimo conjunto de ícones
winamp	xmms	Possui todos os recursos do programa para Windows além de filtros que permite acrescentar efeitos digitais da música (em tempo real), eco, etc.
media player	mplayer,playmidi xwave,	Programas para execução de arquivos de música e videos multimídia. Existem outras alternativas, a escolha depende de seu gosto e da sofisticação do programa.
Agente de Sistema	cron	Pouca diferença. O cron da mais liberdade na programação de tarefas a serem executadas pelo Linux.
Mixer	aumix, cam	Sem diferenças.
Bate-Papo	talk, ytalk	O talk e o ytalk permite a conversa de dois usuários não só através de uma rede local, mas de qualquer parte do planeta, pois usa o protocolo tcp/ip para comunicação. Muito útil e fácil de usar.
MIRC	Bitchx, xchat	Clientes IRC para Linux
IIS, Pers. Web Server	Apache	O apache é o servidor WEB mais usado no mundo (algo em torno de 75% das empresas), muito rápido e flexível de se configurar.
Exchange, NT Mail	Postfix, Sendmail Exim, Qmail	72% da base de servidores de emails no mundo atualmente roda em software livre. Os mais recomenda

são o Postfix e o qmail, devido a segurança, velocidade e integridade de mensagem

Wingate, MS Proxy	Squid, Apache, ip masquerade, nat, diald, exim,	A migração de um servidor proxy para Linux requer o uso de vários programas separados para que se tenha um resultado profissional. Isto pode parecer incomodo no começo, mas você logo perceberá que a divisão de serviços entre programas é mais produtivo. Quando desejar substituir um deles, o funcionamento dos

		outros não serão afetados. Não vou entrar em detalhes sobre os programas citados ao lado, mas o squ é um servidor proxy Web (HTTP e HTTPS) completo e também apresenta u excelente serviço FTP. Possui outros módulos como dns, ping restrições de acesso, limites de tamanho de arquivos, cache, etc.
MS Frontpage	Mozilla e muitas outras ferramentas para geração de conteúdo WEB (como zope, php3, php4, wdm, htdig)	Sem comentários... todas são ferramentas para a geração de grandes Web Sites. O wdm, por exemplo, é usado na geração do site da distribuição Debian (http://www.debian.org) em 30 idiomas diferentes.
MS Winsock	Sem equivalente	O Linux tem suporte nativo a tcp/ip desde o começo de sua existência e não precisa de nenhuma camada de comunicação entre ele e a Internet. A performance é aproximadamente 10% maior em conexões Internet via fax-modem e outras redes tcp/ip.
AVG, Viruscan, norton, F-PROT, CPAV.	Clamavis, AVG F-Prot, ViruScan	Os maiores fabricantes de anti-virus disponibilizam versões para Linux, com o objetivo principal de remoção de vírus em servidores de E-mail ou servidores de arquivos, com o objetivo de não contaminar os vulneráveis sistemas Windows, servindo como uma efetiva barreira de defesa na rede.

Capítulo 4

Discos e Partições

Este capítulo traz explicações de como manipular discos rígidos e partições no sistema GNU/Linux e como acessar seus discos de CD-ROM e partições DOS, Windows 9X/XP/Vista/Seven no GNU/Linux.

4.1 Partições

São divisões existentes no disco rígido que marcam onde começa onde termina um sistema de arquivos. As partições nos permitem usar mais de um sistema operacional no mesmo computador (como o GNU/Linux, Windows e DOS), ou dividir o disco rígido em uma ou mais partes para ser usado por um único sistema operacional ou até mesmo por diferentes arquiteturas (32 e 64 bits).

4.2 Formatando Pen-drives/Disquetes

As subseções seguintes explicarão maneiras de formatar seu pen-drive, memória flash, e outras tecnologias (incluindo disquetes) para serem usados no GNU/Linux e DOS/Windows.

4.2.1 Formatando pen-drives para serem usados no Linux

Para formatar pen-drives para serem usados no GNU/Linux use o comando:

mkfs.ext2 [*-c*] [*/dev/sde1*]

Em alguns sistemas você deve usar mke2fs no lugar de mkfs.ext2. A opção -c faz com que o mkfs.ext2 procure por blocos danificados no pen-drive. Caso deseje formatar um disquete, especifique o dispositivo /dev/fd0 ao inves de /dev/sdb1.

Note que o nome de dispositivo que é conectado varia de acordo com o sistema e quantidade de discos rígidos que sua máquina possui portanto tenha *ATENCÃO* para não formatar

o dispositivo incorreto (que pode ser justamente seu disco disco rígido principal). Para maior segurança, ao identificar o pen-drive, digite `dmesg` ao conectar o pen-drive para visualizar o dispositivo correto ou fique atento as mensagens do console que mostrará o dispositivo que foi associado ao pen-drive.

OBS: Este comando cria um sistema de arquivos *ext2* no pen-drive e permite usar características como permissões de acesso e outras. Isto também faz com que o pen-drive NÃO possa ser lido pelo `DOS/Windows`. Para formatar um pen-drive no `GNU/Linux` usando o *FAT16* ou *FAT32* (compatível com o DOS/Windows) veja próxima seção.

Exemplo: `mkfs.ext2 -c /dev/sde1`

4.2.2 Formatando pen-drives compatíveis com o Windows

A formatação de pen-drives para serem usados no `Windows` é feita usando o comando `mkfs.msdos` que é geralmente incluído no pacote `dosfstools`. O `mkfs.msdos` permite tanto a criação de sistemas de arquivos FAT16 ou FAT32.

`mkfs.msdos [opções] [`*`dispositivo`*`]`

dispositivo Pen-drive que será formatado. Normalmente `/dev/sdb1` (dependendo do dispositivo detectado via comando `dmesg`).

opções

-F [num] Especifica o tipo de FAT que será usado na formatação. Podem ser usados os valores 12 (para formatação usando FAT12, limitado a 12MB), 16 (para formatação usando FAT16, limitado a 2Gb) e 32 (para formatação FAT32, limitado a 128Gb).

-n [nome] Atribui o `[nome]` de volume ao dispositivo.

-c Faz uma pesquisa por bad blocks antes da criação do sistema de arquivos no dispositivo. Os setores defeituosos encontrados serão automaticamente marcados para não serem utilizadas.

Note que não se deve montar o pen-driv / disquete para formata-lo.

Segue abaixo exemplos de como formatar seu pen-drive `mkfs.msdos`:

- `mkfs.msdos /dev/sdc1` - Formata o pen-drive no terceiro dispositivo SCSI Genérico, como FAT32 e usando os valores padrões.
- `mkfs.msdos -F 16 /dev/sdc1` - Faz a mesma coisa que o acima, mas formata o pen-drive como FAT16.
- `mkfs.msdos -n teste -F 16 /dev/sdc1` - Formata o pen-drive no terceiro dispositivo SCSI genérico, como FAT16 e cria o nome de volume `teste`.

4.2.3 Programas de Formatação Gráficos

Além de programas de formatação em modo texto, existem outros para ambiente gráfico (X11) que permitem fazer a mesma tarefa.

Entre os diversos programas destaco o `gfloppy` que além de permitir selecionar se o disquete será formatado para o `GNU/Linux` (ext2), `DOS` (FAT12) e permite selecionar a capacidade e formatação rápida do disco.

4.3 Pontos de Montagem

O GNU/Linux acessa as partições existente em seus discos rígidos e disquetes através de diretórios. Os diretórios que são usados para acessar (montar) partições são chamados de *Pontos de Montagem*. Para detalhes sobre montagem de partições, veja 'Montando (acessando) uma partição de disco' on the next page.

No DOS cada letra de unidade (C:, D:, E:) identifica uma partição de disco, no GNU/Linux os pontos de montagem fazem parte da grande estrutura do sistema de arquivos raiz.

4.4 Identificação de discos e partições em sistemas Linux

No GNU/Linux, os dispositivos existentes em seu computador (como discos rígidos, pendrives, flash, disquetes,, tela, portas de impressora, modem, etc) são identificados por um arquivo referente a este dispositivo no diretório /dev.

A identificação de discos rígidos no GNU/Linux é feita da seguinte forma:

```
/dev/sda1
|    | ||
|    | ||_Número que identifica o número da partição no disco rígido.
|    | |
|    | |_Letra que identifica o disco rígido (a=primeiro, b=segundo, etc...).
|    |
|    |_Sigla que identifica o tipo do disco rígido (sd=SATA/SCSI, sd=IDE, xt=
|
|_Diretório onde são armazenados os dispositivos existentes no sistema.
```

Abaixo algumas identificações de discos e partições em sistemas Linux:

- */dev/fd0* - Primeira unidade de disquetes.
- */dev/fd1* - Segunda unidade de disquetes.
- */dev/sda* - Primeiro disco rígido na primeira controladora SATA ou SCSI.
- */dev/sda1* - Primeira partição do primeiro disco rígido SATA ou.
- */dev/sdb* - Segundo disco rígido na primeira controladora SATA ou SCSI.
- */dev/sdb1* - Primeira partição do segundo disco rígido SATA ou SCSI.
- */dev/sr0* - Primeiro CD-ROM SATA ou SCSI.
- */dev/sr1* - Segundo CD-ROM SATA ou SCSI.

- */dev/hda* - `Primeiro disco rígido na primeira controladora IDE do micro (primary master).`
- */dev/hda1* - `Primeira partição do primeiro disco rígido IDE.`
- */dev/hdb* - `Segundo disco rígido na primeira controladora IDE do micro (primary slave).`
- */dev/hdb1* - `Primeira partição do segundo disco rígido IDE.`
- */dev/xda* - `Primeiro disco rígido XT.`
- */dev/xdb* - `Segundo disco rígido XT.`

As letras de identificação de discos rígidos podem ir além de `sdb`, por exemplo, caso utilize pen-drives, memória flash, as unidades serão detectadas como `sdc`, `sdd` e assim por diante.

É importante entender como os discos e partições são identificados no sistema, pois será necessário usar os parâmetros corretos para monta-los.

4.5 Montando (acessando) uma partição de disco

Você pode acessar uma partição de disco usando o comando `mount`.

`mount [dispositivo] [ponto de montagem] [opções]`

Onde:

dispositivo Identificação da unidade de disco/partição que deseja acessar (como `/dev/hda1` (disco rígido) ou `/dev/fd0` (primeira unidade de disquetes).

ponto de montagem Diretório de onde a *unidade de disco/partição* será acessado. O diretório deve estar vazio para montagem de um sistema de arquivo. Normalmente é usado o diretório `/mnt` para armazenamento de pontos de montagem temporários.

-t [tipo] Tipo do sistema de arquivos usado pelo *dispositivo*. São aceitos os sistemas de arquivos:

- *ext2* - Para partições `GNU/Linux` usando o Extended File System versão 2 (a mais comum).
- *ext3* - Para partições `GNU/Linux` usando o Extended File System versão 3, com suporte a journaling.
- *ext4* - Para partições `GNU/Linux` usando o Extended File System versão 4, com suporte a journaling.
- *reiserfs* - Para partições reiserfs, com suporte a journaling.
- *xfs* - Para partições xfs, com suporte a journaling.
- *vfat* - Para partições `Windows 95` que utilizam nomes extensos de arquivos e diretórios.
- *msdos* - Para partições `DOS` normais.
- *iso9660* - Para montar unidades de `CD-ROM`. É o padrão.

Na maioria das vezes, caso o sistema de arquivos não seja especificado, o `mount` utilizará a auto-detecção e montará a partição usando o sistema de arquivos correto. Para mais detalhes sobre opções usadas com cada sistema de arquivos, veja a página de manual *mount*.

-r Caso for especificada, monta a partição somente para leitura.

-w Caso for especificada, monta a partição como leitura/gravação. É o padrão.

Existem muitas outras opções que podem ser usadas com o comando `mount`, mas aqui procurei somente mostrar o básico para "montar" seus discos e partições no `GNU/Linux` (para mais opções, veja a página de manual do `mount`). Caso você digitar `mount` sem parâmetros, serão mostrados os sistemas de arquivos atualmente montados no sistema. Esta mesma listagem pode ser vista em `/etc/mtab`. A remontagem de partição também é muito útil, especialmente após reparos nos sistema de arquivos do disco rígido. Veja alguns exemplos de remontagem abaixo.

É necessário permissões de root para montar partições, a não ser que tenha especificado a opção `user` no arquivo `/etc/fstab` (veja 'fstab' on this page).

Exemplo de Montagem:

- Montar uma partição Windows (vfat) de `/dev/sda1` em `/mnt` somente para leitura: `mount /dev/sda1 /mnt -r -t vfat`
- Montar um pen-drive detectado em `/dev/sdc1` em `/mnt`: `mount /dev/sdc1 /mnt -t vfat`
- Montar uma partição DOS localizada em um segundo disco rígido `/dev/hdb1` em `/mnt`: `mount /dev/hdb1 /mnt -t msdos`.
- Remontar a partição raíz como somente leitura: `mount -o remount,ro /`
- Remontar a partição raíz como *leitura/gravação* (a opção -n é usada porque o `mount` não conseguirá atualizar o arquivo `/etc/mtab` devido ao sistema de arquivos / estar montado como somente leitura atualmente: `mount -n -o remount,rw /`.

4.5.1 fstab

O arquivo `/etc/fstab` permite que as partições do sistema sejam montadas facilmente especificando somente o dispositivo ou o ponto de montagem. Este arquivo contém parâmetros sobre as partições que são lidos pelo comando `mount`. Cada linha deste arquivo contém a partição que desejamos montar, o ponto de montagem, o sistema de arquivos usado pela partição e outras opções. `fstab` tem a seguinte forma:

```
Sistema_de_arquivos Ponto_de_Montagem Tipo    Opções            dump ordem
/dev/sda1           /                 ext3    defaults             0    1
/dev/sda2           /boot             ext3    defaults             0    2
/dev/sda3           /dos              msdos   defaults,noauto,rw  0    0
/dev/hdg            /cdrom            iso9660 defaults,noauto     0    0
```

Onde:

Sistema de Arquivos Partição que deseja montar.

Ponto de montagem Diretório do `GNU/Linux` onde a partição montada será acessada.

Tipo Tipo de sistema de arquivos usado na partição que será montada. Para partições `GNU/Linux` use *ext3*, *reiserfs*, *xfs* (de acordo com o tipo de partição selecionada durante a formatação), para partições `DOS` (sem nomes extensos de arquivos) use *msdos*, para partições `Win 95` (com suporte a nomes extensos de arquivos) use *vfat*, para unidades de CD-ROM use *iso9660*.

Opções Especifica as opções usadas com o sistema de arquivos. Abaixo, algumas opções de montagem para ext2/3/4 (a lista completa pode ser encontrada na página de manual do `mount`):

- `defaults` - Utiliza valores padrões de montagem.
- `noauto` - Não monta os sistemas de arquivos durante a inicialização (útil para CD-ROMS e disquetes).
- `ro` - Monta como somente leitura.
- `user` - Permite que usuários montem o sistema de arquivos (não recomendado por motivos de segurança).
- `sync` é recomendado para uso com discos removíveis (disquetes, zip drives, nfs, etc) para que os dados sejam gravados imediatamente na unidade (caso não seja usada, você deve usar o comando 'sync' on page 78 antes de retirar o disquete da unidade.

dump Especifica a frequência de backup feita com o programa `dump` no sistema de arquivos. 0 desativa o backup.

Ordem Define a ordem que os sistemas de arquivos serão verificados na inicialização do sistema. Se usar 0, o sistema de arquivos não é verificado. O sistema de arquivos raíz que deverá ser verificado primeiro é o raíz "/" .

Após configurar o `/etc/fstab`, basta digitar o comando `mount /dev/hdg` ou `mount /cdrom` para que a unidade de CD-ROM seja montada. Você deve ter notado que não é necessário especificar o sistema de arquivos da partição pois o `mount` verificará se ele já existe no `/etc/fstab` e caso existir, usará as opções especificadas neste arquivo. Para maiores detalhes veja as páginas de manual `fstab` e `mount`.

4.6 Desmontando uma partição de disco

Utilize o comando `umount` para desmontar um sistema de arquivos que foi montado com o `mount`. Você deve ter permissões de root para desmontar uma partição.

`umount [`*`dispositivo/ponto de montagem`*`]`

Você pode tanto usar `umount /dev/sda1` como `umount /mnt` para desmontar um sistema de arquivos `/dev/sda1` montado em `/mnt`.

Observação: O comando `umount` executa o `sync` automaticamente no momento da desmontagem, para garantir que todos os dados ainda em memória RAM sejam salvos.

Capítulo 5

Execução de programas

Este capítulo explica como executar programas no `GNU/Linux` e o uso das ferramentas de controle de execução dos programas.

5.1 Executando um comando/programa

Para executar um comando, é necessário que ele tenha permissões de execução (veja 'Tipos de Permissões de Acesso' on page 94 e 'ls' on page 61) e que esteja no caminho de procura de arquivos (veja 'path' on this page).

No aviso de comando #(root) ou $(usuário), digite o nome do comando e tecle Enter. O programa/comando é executado e receberá um número de identificação (chamado de PID - Process Identification), este número é útil para identificar o processo no sistema e assim ter um controle sobre sua execução (será visto mais adiante neste capítulo).

Todo o programa executado no `GNU/Linux` roda sob o controle das permissões de acesso. Recomendo ver mais tarde o 'Permissões de acesso a arquivos e diretórios' on page 93.

Exemplos de comandos: `ls`, `df`, `pwd`.

5.2 path

Path é o caminho de procura dos arquivos/comandos executáveis. O path (caminho) é armazenado na variável de ambiente `PATH`. Você pode ver o conteúdo desta variável com o comando `echo $PATH`.

Por exemplo, o caminho `/usr/local/bin:/usr/bin:/bin:/usr/bin/X11` significa que se você digitar o comando `ls`, o interpretador de comandos iniciará a procura do programa `ls` no diretório `/usr/local/bin`, caso não encontre o arquivo no diretório `/usr/local/bin` ele inicia a procura em `/usr/bin`, até que encontre o arquivo procurado.

Caso o interpretador de comandos chegue até o último diretório do path e não encontre o arquivo/comando digitado, é mostrada a seguinte mensagem:

`bash: ls: command not found` (comando não encontrado).

O caminho de diretórios vem configurado na instalação do Linux, mas pode ser alterado no arquivo `/etc/profile`. Caso deseje alterar o caminho para todos os usuários, este arquivo é o melhor lugar, pois ele é lido por todos os usuários no momento do login.

Caso um arquivo/comando não esteja localizado em nenhum dos diretórios do path, você deve executa-lo usando um `./` na frente do comando.

Se deseja alterar o `path` para um único usuário, modifique o arquivo `.bash_profile` em seu diretório de usuário (home).

OBSERVAÇÃO: Por motivos de segurança, não inclua o diretório atual `$PWD` no `path`.

5.3 Tipos de Execução de comandos/programas

Um programa pode ser executado de duas formas:

1. `Primeiro Plano` - Também chamado de *foreground*. Quando você deve esperar o término da execução de um programa para executar um novo comando. Somente é mostrado o aviso de comando após o término de execução do comando/programa.
2. `Segundo Plano` - Também chamado de *background*. Quando você não precisa esperar o término da execução de um programa para executar um novo comando. Após iniciar um programa em *background*, é mostrado um número PID (identificação do Processo) e o aviso de comando é novamente mostrado, permitindo o uso normal do sistema. O programa executado em background continua sendo executado internamente. Após ser concluído, o sistema retorna uma mensagem de pronto acompanhado do número PID do processo que terminou.

Para iniciar um programa em `primeiro plano`, basta digitar seu nome normalmente. Para iniciar um programa em `segundo plano`, acrescente o caracter "`&`" após o final do comando.

OBS: Mesmo que um usuário execute um programa em segundo plano e saia do sistema, o programa continuará sendo executado até que seja concluído ou finalizado pelo usuário que iniciou a execução (ou pelo usuário root).

Exemplo: `find / -name boot.b &`

O comando será executado em segundo plano e deixará o sistema livre para outras tarefas. Após o comando `find` terminar, será mostrada uma mensagem.

5.4 Executando programas em seqüência

Os comandos podem ser executados em seqüência (um após o término do outro) se os separarmos com ";". Por exemplo: `echo primeiro;echo segundo;echo terceiro`

5.5 ps

Algumas vezes é útil ver quais processos estão sendo executados no computador. O comando `ps` faz isto, e também nos mostra qual usuário executou o programa, hora que o processo foi iniciado, etc.

`ps [opções]`

Onde:
opções
a Mostra os processos criados por você e de outros usuários do sistema.
x Mostra processos que não são controlados pelo terminal.
u Mostra o nome de usuário que iniciou o processo e hora em que o processo foi iniciado.
m Mostra a memória ocupada por cada processo em execução.
f Mostra a árvore de execução de comandos (comandos que são chamados por outros comandos).
e Mostra variáveis de ambiente no momento da inicialização do processo.
w Mostra a continuação da linha atual na próxima linha ao invés de cortar o restante que não couber na tela.
As opções acima podem ser combinadas para resultar em uma listagem mais completa. Você também pode usar pipes "`|`" para `filtrar` a saída do comando `ps`. Para detalhes, veja '| (pipe)' on page 106.

Ao contrário de outros comandos, o comando `ps` não precisa do hífen "-" para especificar os comandos. Isto porque ele não utiliza opções longas e não usa parâmetros.

Exemplos: `ps`, `ps ax|grep inetd`, `ps auxf`, `ps auxw`.

5.6 top

Mostra os programas em execução ativos, parados, tempo usado na CPU, detalhes sobre o uso da memória RAM, Swap, disponibilidade para execução de programas no sistema, etc.

`top` é um programa que continua em execução mostrando continuamente os processos que estão rodando em seu computador e os recursos utilizados por eles. Para sair do `top`, pressione a tecla `q`.

`top [opções]`

Onde:
-d [tempo] Atualiza a tela após o [tempo] (em segundos).
-s Diz ao `top` para ser executado em modo seguro.
-i Inicia o `top` ignorando o tempo de processos zumbis.
-c Mostra a linha de comando ao invés do nome do programa.
A ajuda sobre o `top` pode ser obtida dentro do programa pressionando a tecla `h` ou pela página de manual (`man top`).

Abaixo algumas teclas úteis:

- `espaço` - Atualiza imediatamente a tela.
- `CTRL+L` - Apaga e atualiza a tela.
- `h` - Mostra a tela de ajuda do programa. É mostrado todas as teclas que podem ser usadas com o `top`.
- `i` - Ignora o tempo ocioso de processos zumbis.
- `q` - Sai do programa.
- `k` - Finaliza um processo - semelhante ao comando `kill`. Você será perguntado pelo número de identificação do processo (PID). Este comando não estará disponível caso esteja usando o `top` com a opção `-s`.
- `n` - Muda o número de linhas mostradas na tela. Se 0 for especificado, será usada toda a tela para listagem de processos.

5.7 Controle de execução de processos

Abaixo algumas comandos e métodos úteis para o controle da execução de processos no `GNU/Linux`.

5.7.1 Interrompendo a execução de um processo

Para cancelar a execução de algum processo `rodando em primeiro plano`, basta pressionar as teclas `CTRL+C`. A execução do programa será cancelada e será mostrado o aviso de comando. Você também pode usar o comando 'kill' on the next page para interromper um processo sendo executado.

5.7.2 Parando momentaneamente a execução de um processo

Para parar a execução de um processo rodando em primeiro plano, basta pressionar as teclas `CTRL+Z`. O programa em execução será pausado e será mostrado o número de seu job e o aviso de comando.

Para retornar a execução de um comando pausado, use 'fg' on the facing page ou 'bg' on the next page.

O programa permanece na memória no ponto de processamento em que parou quando ele é interrompido. Você pode usar outros comandos ou rodar outros programas enquanto o programa atual está interrompido.

5.7.3 jobs

O comando `jobs` mostra os processos que estão parados ou rodando em *segundo plano*. Processos em segundo plano são iniciados usando o símbolo "`&`" no final da linha de comando (veja 'Tipos de Execução de comandos/programas' on page 52) ou através do comando `bg`.

`jobs`

O número de identificação de cada processo parado ou em segundo plano (job), é usado com os comandos 'fg' on the current page e 'bg' on this page. Um processo interrompido pode ser finalizado usando-se o comando `kill %[num]`, onde `[num]` é o número do processo obtido pelo comando `jobs`.

5.7.4 fg

Permite fazer um programa rodando em segundo plano ou parado, rodar em primeiro plano. Você deve usar o comando `jobs` para pegar o número do processo rodando em segundo plano ou interrompida, este número será passado ao comando `fg` para ativa-lo em primeiro plano.

`fg [`*`número`*`]`

Onde *número* é o número obtido através do comando `jobs`.

Caso seja usado sem parâmetros, o `fg` utilizará o último programa interrompido (o maior número obtido com o comando `jobs`).

Exemplo: `fg 1`.

5.7.5 bg

Permite fazer um programa rodando em primeiro plano ou parado, rodar em segundo plano. Para fazer um programa em primeiro plano rodar em segundo, é necessário primeiro interromper a execução do comando com `CTRL+ Z`, será mostrado o número da tarefa interrompida, use este número com o comando `bg` para iniciar a execução do comando em segundo plano.

`bg [`*`número`*`]`

Onde: *número* número do programa obtido com o pressionamento das teclas `CTRL+Z` ou através do comando `jobs`.

5.7.6 kill

Permite enviar um sinal a um comando/programa. Caso seja usado sem parâmetros, o `kill` enviará um sinal de término ao processo sendo executado.

`kill [`*`opções`*`] [`*`sinal`*`] [`*`número`*`]`

Onde:

número É o número de identificação do processo obtido com o comando 'ps' on page 53. Também pode ser o número após o sinal de `%` obtido pelo comando `jobs` para matar uma tarefa interrompida. Veja 'jobs' on the preceding page.

sinal Sinal que será enviado ao processo. Se omitido usa `-15` como padrão.

opções

-9 Envia um sinal de destruição ao processo ou programa. Ele é terminado imediatamente sem chances de salvar os dados ou apagar os arquivos temporários criados por ele.

Você precisa ser o dono do processo ou o usuário root para termina-lo ou destruí-lo. Você pode verificar se o processo foi finalizado através do comando `ps`. Os tipos de sinais aceitos pelo `GNU/Linux` são explicados em detalhes em 'Sinais do Sistema' on the current page.

Exemplo: `kill 500`, `kill -9 500`, `kill %1`.

5.7.7 killall

Permite finalizar processos através do nome.

`killall [opções] [sinal] [processo]`

Onde:

processo Nome do processo que deseja finalizar

sinal Sinal que será enviado ao processo (pode ser obtido usando a opção `-i`).

opções

-i Pede confirmação sobre a finalização do processo.

-l Lista o nome de todos os sinais conhecidos.

-q Ignora a existência do processo.

-v Retorna se o sinal foi enviado com sucesso ao processo.

-w Finaliza a execução do `killall` somente após finalizar todos os processos.

Os tipos de sinais aceitos pelo `GNU/Linux` são explicados em detalhes na 'Sinais do Sistema' on this page.

Exemplo: `killall -HUP inetd`

5.7.8 killall5

Envia um sinal de finalização para todos os processos sendo executados.

`killall5 [sinal]`

5.7.9 Sinais do Sistema

Retirado da página de manual `signal`. O `GNU/Linux` suporta os sinais listados abaixo. Alguns números de sinais são dependentes de arquitetura.

Primeiro, os sinais descritos no *POSIX 1*:

Sinal	Valor	Ação	Comentário
HUP	1	A	Travamento detectado no terminal de controle ou finalização do processo controlado
INT	2	A	Interrupção através do teclado
QUIT	3	C	Sair através do teclado
ILL	4	C	Instrução Ilegal
ABRT	6	C	Sinal de abortar enviado pela função abort
FPE	8	C	Exceção de ponto Flutuante
KILL	9	AEF	Sinal de destruição do processo
SEGV	11	C	Referência Inválida de memória
PIPE	13	A	Pipe Quebrado: escreveu para o pipe sem leitores
ALRM	14	A	Sinal do Temporizador da chamada do sistema alarm
TERM	15	A	Sinal de Término
USR1	30,10,16	A	Sinal definido pelo usuário 1
USR2	31,12,17	A	Sinal definido pelo usuário 2
CHLD	20,17,18	B	Processo filho parado ou terminado
CONT	19,18,25		Continuar a execução, se interrompido
STOP	17,19,23	DEF	Interromper processo
TSTP	18,20,24	D	Interromper digitação no terminal
TTIN	21,21,26	D	Entrada do terminal para o processo em segundo pla
TTOU	22,22,27	D	Saída do terminal para o processo em segundo plano

As letras da coluna `Ação` tem o seguinte significado:

- `A` - A ação padrão é terminar o processo.
- `B` - A ação padrão é ignorar o sinal.
- `C` - A ação padrão é terminar o processo e mostrar o core.
- `D` - A ação padrão é parar o processo.
- `E` - O sinal não pode ser pego.
- `F` - O sinal não pode ser ignorado.

Sinais não descritos no *POSIX 1* mas descritos na *SUSv2*:

Sinal	Valor	Ação	Comentário
BUS	10,7,10	C	Erro no Barramento (acesso incorreto da memória)
POLL		A	Evento executado em Pool (Sys V). Sinônimo de IO
PROF	27,27,29	A	Tempo expirado do Profiling
SYS	12,-,12	C	Argumento inválido para a rotina (SVID)
TRAP	5	C	Captura do traço/ponto de interrupção
URG	16,23,21	B	Condição Urgente no soquete (4.2 BSD)
VTALRM	26,26,28	A	Alarme virtual do relógio (4.2 BSD)
XCPU	24,24,30	C	Tempo limite da CPU excedido (4.2 BSD)
XFSZ	25,25,31	C	Limite do tamanho de arquivo excedido (4.2 BSD)

(Para os casos SIGSYS, SIGXCPU, SIGXFSZ, e em algumas arquiteturas também o SIGGUS, a

ação padrão do Linux para kernels 2.3.27 e superiores é A (terminar), enquanto SYSv2 descreve C (terminar e mostrar dump core).) Seguem vários outros sinais:

Sinal	Valor	Ação	Comentário
IOT	6	C	Traço IOT. Um sinônimo para ABRT
EMT	7,-,7		
STKFLT	-,16,-	A	Falha na pilha do processador
IO	23,29,22	A	I/O agora possível (4.2 BSD)
CLD	-,-,18		Um sinônimo para CHLD
PWR	29,30,19	A	Falha de força (System V)
INFO	29,-,-		Um sinônimo para SIGPWR
LOST	-,-,-	A	Perda do bloqueio do arquivo
WINCH	28,28,20	B	Sinal de redimensionamento da Janela (4.3 BSD, S
UNUSED	-,31,-	A	Sinal não usado (será SYS)

O "-" significa que o sinal não está presente. Onde três valores são listados, o primeiro é normalmente válido para o Alpha e Sparc, o do meio para i386, PowerPc e sh, o último para o Mips. O sinal 29 é SIGINFO/SIGPWR em um Alpha mas SIGLOST em um Sparc.

5.8 Fechando um programa quando não se sabe como sair

Muitas vezes quando se esta iniciando no `GNU/Linux` você pode executar um programa e talvez não saber como fecha-lo. Este capítulo do guia pretende ajuda-lo a resolver este tipo de problema.

Isto pode também ocorrer com programadores que estão construindo seus programas e por algum motivo não implementam uma opção de saída, ou ela não funciona!

Em nosso exemplo vou supor que executamos um programa em desenvolvimento com o nome `contagem` que conta o tempo em segundos a partir do momento que é executado, mas que o programador esqueceu de colocar uma opção de saída. Siga estas dicas para finaliza-lo:

1 Normalmente todos os programas `UNIX` (o `GNU/Linux` também é um Sistema Operacional baseado no `UNIX`) podem ser interrompidos com o pressionamento das teclas `<CTRL>` e `<C>`. Tente isto primeiro para finalizar um programa. Isto provavelmente não vai funcionar se estiver usando um Editor de Texto (ele vai entender como um comando de menu). Isto normalmente funciona para comandos que são executados e terminados sem a intervenção do usuário.

  Caso isto não der certo, vamos partir para a força! ;-)

2 Mude para um novo console (pressionando `<ALT>` e `<F2>`), e faça o *login* como usuário **root**.

3 Localize o PID (número de identificação do processo) usando o comando: `ps ax`, aparecerão várias linhas cada uma com o número do processo na primeira coluna, e a linha de comando do programa na última coluna. Caso aparecerem vários processos você pode usar `ps ax|grep contagem`, neste caso o `grep` fará uma filtragem da saída do comando `ps ax` mostrando somente as linhas que tem a palavra "contagem". Para maiores detalhes, veja o comando 'grep' on page 73.

4 Feche o processo usando o comando `kill` *`PID`*, lembre-se de substituir PID pelo número encontrado pelo comando `ps ax` acima.

   O comando acima envia um sinal de término de execução para o processo (neste caso o programa `contagem`). O sinal de término mantém a chance do programa salvar seus dados ou apagar os arquivos temporários que criou e então ser finalizado, isto depende do programa.

5 Alterne para o console onde estava executando o programa `contagem` e verifique se ele ainda está em execução. Se ele estiver parado mas o aviso de comando não está disponível, pressione a tecla <ENTER>. Freqüentemente acontece isto com o comando `kill`, você finaliza um programa mas o aviso de comando não é mostrado até que se pressione <ENTER>.

6 Caso o programa ainda não foi finalizado, repita o comando `kill` usando a opção -9: `kill -9 PID`. Este comando envia um sinal de DESTRUIÇÃO do processo, fazendo ele terminar "na marra"!

Uma última dica: todos os programas estáveis (todos que acompanham as boas distribuições `GNU/Linux`) tem sua opção de saída. Lembre-se que quando finaliza um processo todos os dados do programa em execução podem ser perdidos (principalmente se estiver em um editor de textos), mesmo usando o `kill` sem o parâmetro `-9`.

Procure a opção de saída de um programa consultando o help on line, as páginas de manual, a documentação que acompanha o programa, info pages. Para detalhes de como encontrar a ajuda dos programas, veja o 'Como obter ajuda no sistema' on page 119

5.9 Eliminando caracteres estranhos

As vezes quando um programa `mal comportado` é finalizado ou quando você visualiza um arquivo binário através do comando `cat`, é possível que o aviso de comando (prompt) volte com caracteres estranhos.

Para fazer tudo voltar ao normal, basta digitar `reset` e teclar `ENTER`. Não se preocupe, o comando `reset` não reiniciará seu computador (como o botão reset do seu computador faz), ele apenas fará tudo voltar ao normal.

Note que enquanto você digitar `reset` aparecerão caracteres estranhos ao invés das letras. Não se preocupe! Basta digitar corretamente e bater `ENTER` e o aviso de comando voltará ao normal.

Capítulo 6

Comandos para manipulação de diretório

Abaixo comandos úteis para a manipulação de diretórios.

6.1 ls

Lista os arquivos de um diretório.

`ls [opções] [caminho/arquivo] [caminho1/arquivo1] ...`

onde:

caminho/arquivo Diretório/arquivo que será listado.

caminho1/arquivo1 Outro Diretório/arquivo que será listado. Podem ser feitas várias listagens de uma só vez.

opções

-a, –all Lista todos os arquivos (inclusive os ocultos) de um diretório.

-A, –almost-all Lista todos os arquivos (inclusive os ocultos) de um diretório, exceto o diretório atual e o de nível anterior.

-B, –ignore-backups Não lista arquivos que terminam com ~ (Backup).

–color=PARAM Mostra os arquivos em cores diferentes, conforme o tipo de arquivo. PARAM pode ser:

- *never* - Nunca lista em cores (mesma coisa de não usar o parâmetro –color).
- *always* - Sempre lista em cores conforme o tipo de arquivo.
- *auto* - Somente colore a listagem se estiver em um terminal.

-d, –directory Lista os nomes dos diretórios ao invés do conteúdo.

-f Não classifica a listagem.

-F Insere um caracter após arquivos executáveis ('*'), diretórios ('/'), soquete ('='), link simbólico ('@') e pipe ('|'). Seu uso é útil para identificar de forma fácil tipos de arquivos nas listagens de diretórios.

-G, –no-group Oculta a coluna de grupo do arquivo.

-h, –human-readable Mostra o tamanho dos arquivos em Kbytes, Mbytes, Gbytes.

-H Faz o mesmo que `-h`, mas usa unidades de 1000 ao invés de 1024 para especificar Kbytes, Mbytes, Gbytes.
-l Usa o formato longo para listagem de arquivos. Lista as permissões, data de modificação, donos, grupos, etc.
-n Usa a identificação de usuário e grupo numérica ao invés dos nomes.
-L, –dereference Lista o arquivo original e não o link referente ao arquivo.
-o Usa a listagem longa sem os donos dos arquivos (mesma coisa que -lG).
-p Mesma coisa que -F, mas não inclui o símbolo '*' em arquivos executáveis. Esta opção é típica de sistemas `Linux`.
-R Lista diretórios e sub-diretórios recursivamente.
Uma listagem feita com o comando `ls -la` normalmente é mostrada da seguinte maneira:

```
-rwxr-xr--  1  gleydson user    8192 nov 4 16:00 teste
```

Abaixo as explicações de cada parte:
`-rwxr-xr--` São as permissões de acesso ao arquivo teste. A primeira letra (da esquerda) identifica o tipo do arquivo, se tiver um `d` é um diretório, se tiver um "-" é um arquivo normal. As permissões de acesso é explicada em detalhes em 'Permissões de acesso a arquivos e diretórios' on page 93.
`1` Se for um diretório, mostra a quantidade de sub-diretórios existentes dentro dele. Caso for um arquivo, será 1.
`gleydson` Nome do dono do arquivo `teste`.
`user` Nome do grupo que o arquivo `teste` pertence.
`8192` Tamanho do arquivo (em bytes).
`nov` Mês da criação/ última modificação do arquivo.
`4` Dia que o arquivo foi criado.
`16:00` Hora em que o arquivo foi criado/modificado. Se o arquivo foi criado há mais de um ano, em seu lugar é mostrado o ano da criação do arquivo.
`teste` Nome do arquivo.

Exemplos do uso do comando `ls`:
- `ls` - Lista os arquivos do diretório atual.
- `ls /bin /sbin` - Lista os arquivos do diretório /bin e /sbin
- `ls -la /bin` - Listagem completa (vertical) dos arquivos do diretório /bin inclusive os ocultos.

6.2 cd

Entra em um diretório. Você precisa ter a permissão de execução para entrar no diretório.

`cd [`*`diretório`*`]`

onde:

diretório - diretório que deseja entrar.

Exemplos:

- Usando `cd` sem parâmetros ou `cd ~`, você retornará ao seu diretório de usuário (diretório home).
- `cd /`, retornará ao diretório raíz.
- `cd -`, retornará ao diretório anteriormente acessado.
- `cd ..`, sobe um diretório.
- `cd ../[diretório]`, sobe um diretório e entra imediatamente no próximo (por exemplo, quando você está em `/usr/sbin`, você digita `cd ../bin`, o comando `cd` retorna um diretório (`/usr`) e entra imediatamente no diretório `bin` (`/usr/bin`).

6.3 pwd

Mostra o nome e caminho do diretório atual.

Você pode usar o comando pwd para verificar em qual diretório se encontra (caso seu aviso de comandos não mostre isso).

6.4 mkdir

Cria um diretório no sistema. Um diretório é usado para armazenar arquivos de um determinado tipo. O diretório pode ser entendido como uma *pasta* onde você guarda seus papeis (arquivos). Como uma pessoa organizada, você utilizará uma pasta para guardar cada tipo de documento, da mesma forma você pode criar um diretório `vendas` para guardar seus arquivos relacionados com vendas naquele local.

`mkdir [opções] [caminho/diretório] [caminho1/diretório1]`

onde:
caminho Caminho onde o diretório será criado.
diretório Nome do diretório que será criado.
opções:
-p Caso os diretórios dos níveis acima não existam, eles também serão criados.
–verbose Mostra uma mensagem para cada diretório criado. As mensagens de erro serão mostradas mesmo que esta opção não seja usada.
Para criar um novo diretório, você deve ter permissão de gravação. Por exemplo, para criar um diretório em /tmp com o nome de `teste` que será usado para gravar arquivos de teste, você deve usar o comando "`mkdir/tmp/teste`".

Podem ser criados mais de um diretório com um único comando (`mkdir /tmp/teste /tmp/teste1 /tmp/teste2`).

6.5 rmdir

Remove um diretório do sistema. Este comando faz exatamente o contrário do `mkdir`. O diretório a ser removido deve estar vazio e você deve ter permissão de gravação para remove-

lo.

`rmdir [caminho/diretório] [caminho1/diretório1]`

onde:
caminho Caminho do diretório que será removido.
diretório Nome do diretório que será removido.
É necessário que esteja um nível acima do diretório(s) que será(ão) removido(s). Para remover diretórios que contenham arquivos, use o comando `rm` com a opção `-r` (para maiores detalhes, veja 'rm' on page 66).

Por exemplo, para remover o diretório `/tmp/teste` você deve estar no diretório `tmp` e executar o comando `rmdir teste`.

Capítulo 7

Comandos para manipulação de Arquivos

Abaixo, comandos utilizados para manipulação de arquivos.

7.1 cat

Mostra o conteúdo de um arquivo binário ou texto.

`cat [opções] [`*`diretório/arquivo`*`] [`*`diretório1/arquivo1`*`]`
diretório/arquivo Localização do arquivo que deseja visualizar o conteúdo.
opções
-n, –number Mostra o número das linhas enquanto o conteúdo do arquivo é mostrado.
-s, –squeeze-blank Não mostra mais que uma linha em branco entre um parágrafo e outro.
- Lê a entrada padrão.
O comando `cat` trabalha com arquivos texto. Use o comando `zcat` para ver diretamente arquivos compactados com `gzip`.

Exemplo: `cat /usr/doc/copyright/GPL`

7.2 tac

Mostra o conteúdo de um arquivo binário ou texto (como o `cat`) só que em ordem inversa.

`tac [opções] [`*`diretório/arquivo`*`] [`*`diretório1/arquivo1`*`]`
diretório/arquivo Localização do arquivo que deseja visualizar o conteúdo
opções
-s [string] Usa o [string] como separador de registros.
- Lê a entrada padrão.

Exemplo: `tac /usr/doc/copyright/GPL`.

7.3 rm

Apaga arquivos. Também pode ser usado para apagar diretórios e sub-diretórios vazios ou que contenham arquivos.

`rm [opções][caminho][arquivo/diretório] [caminho1][arquivo1/diretório1]`

onde:

caminho Localização do arquivo que deseja apagar. Se omitido, assume que o arquivo esteja no diretório atual.

arquivo/diretório Arquivo que será apagado.

opções

-i, –interactive Pergunta antes de remover, esta é ativada por padrão.

-v, –verbose Mostra os arquivos na medida que são removidos.

-r, –recursive Usado para remover arquivos em sub-diretórios. Esta opção também pode ser usada para remover sub-diretórios.

-f, –force Remove os arquivos sem perguntar.

– arquivo Remove arquivos/diretórios que contém caracteres especiais. O separador "–" funciona com todos os comandos do shell e permite que os caracteres especiais como "*", "?", "-", etc. sejam interpretados como caracteres comuns.

Use com atenção o comando `rm`, uma vez que os arquivos e diretórios forem apagados, eles não poderão ser mais recuperados.

Exemplos:

- `rm teste.txt` - Apaga o arquivo `teste.txt` no diretório atual.
- `rm *.txt` - Apaga todos os arquivos do diretório atual que terminam com `.txt`.
- `rm *.txt teste.novo` - Apaga todos os arquivos do diretório atual que terminam com `.txt` e também o arquivo `teste.novo`.
- `rm -rf /tmp/teste/*` - Apaga todos os arquivos e sub-diretórios do diretório `/tmp/teste` mas mantém o sub-diretório `/tmp/teste`.
- `rm -rf /tmp/teste` - Apaga todos os arquivos e sub-diretórios do diretório `/tmp/teste`, inclusive `/tmp/teste`.
- `rm -f -- --arquivo--` - Remove o arquivo de nome `–arquivo–`.

7.4 cp

Copia arquivos.

`cp [opções] [origem] [destino]`

onde:

origem Arquivo que será copiado. Podem ser especificados mais de um arquivo para ser copiado usando "Curingas" (veja 'Curingas' on page 33).

destino O caminho ou nome de arquivo onde será copiado. Se o destino for um diretório, os arquivos de origem serão copiados para dentro do diretório.

opções

i, –interactive Pergunta antes de substituir um arquivo existente.

-f, –force Não pergunta, substitui todos os arquivos caso já exista.
-r Copia arquivos dos diretórios e subdiretórios da origem para o destino. É recomendável usar -R ao invés de -r.
-R, –recursive Copia arquivos e sub-diretórios (como a opção -r) e também os arquivos especiais FIFO e dispositivos.
-v, –verbose Mostra os arquivos enquanto estão sendo copiados.
O comando `cp` copia arquivos da ORIGEM para o DESTINO. Ambos origem e destino terão o mesmo conteúdo após a cópia.

Exemplos:
`cp teste.txt teste1.txt` Copia o arquivo `teste.txt` para `teste1.txt`.
`cp teste.txt /tmp` Copia o arquivo `teste.txt` para dentro do diretório `/tmp`.
`cp * /tmp` Copia todos os arquivos do diretório atual para `/tmp`.
`cp /bin/* .` Copia todos os arquivos do diretório `/bin` para o diretório em que nos encontramos no momento.
`cp -R /bin /tmp` Copia o diretório `/bin` e todos os arquivos/sub-diretórios existentes para o diretório `/tmp`.
`cp -R /bin/* /tmp` Copia todos os arquivos do diretório `/bin` (exceto o diretório `/bin`) e todos os arquivos/sub-diretórios existentes dentro dele para `/tmp`.
`cp -R /bin /tmp` Copia todos os arquivos e o diretório `/bin` para `/tmp`.

7.5 mv

Move ou renomeia arquivos e diretórios. O processo é semelhante ao do comando `cp` mas o arquivo de origem é apagado após o término da cópia.

`mv [`*`opções`*`] [`*`origem`*`] [`*`destino`*`]`

Onde:
origem Arquivo/diretório de origem.
destino Local onde será movido ou novo nome do arquivo/diretório.
opções
-f, –force Substitui o arquivo de destino sem perguntar.
-i, –interactive Pergunta antes de substituir. É o padrão.
-v, –verbose Mostra os arquivos que estão sendo movidos.
O comando `mv` copia um arquivo da *ORIGEM* para o *DESTINO* (semelhante ao `cp`), mas após a cópia, o arquivo de *ORIGEM* é apagado.

Exemplos:
`mv teste.txt teste1.txt` Muda o nome do arquivo `teste.txt` para `teste1.txt`.
`mv teste.txt /tmp` Move o arquivo teste.txt para `/tmp`. Lembre-se que o arquivo de origem é apagado após ser movido.
`mv teste.txt teste.new` (supondo que `teste.new` já exista) Copia o arquivo `teste.txt` por cima de `teste.new` e apaga `teste.txt` após terminar a cópia.

Capítulo 8

Comandos Diversos

Comandos de uso diversos no sistema.

8.1 clear

Limpa a tela e posiciona o cursor no canto superior esquerdo do vídeo.

`clear`

8.2 date

Permite ver/modificar a Data e Hora do Sistema. Você precisa estar como usuário root para modificar a data e hora. .

`date MesDiaHoraMinuto[AnoSegundos]`

Onde:

MesDiaHoraMinuto[AnoSegundos] São respectivamente os números do mês, dia, hora e minutos sem espaços. Opcionalmente você pode especificar o Ano (com 2 ou 4 dígitos) e os Segundos.

+[FORMATO] Define o formato da listagem que será usada pelo comando `date`. Os seguintes formatos são os mais usados:

- `%d` - Dia do Mês (00-31).
- `%m` - Mês do Ano (00-12).
- `%y` - Ano (dois dígitos).
- `%Y` - Ano (quatro dígitos).
- `%H` - Hora (00-24).
- `%I` - Hora (00-12).
- `%M` - Minuto (00-59).
- `%j` - Dia do ano (1-366).
- `%p` - AM/PM (útil se utilizado com %d).

- `%r` - Formato de 12 horas completo (hh:mm:ss AM/PM).
- `%T` - Formato de 24 horas completo (hh:mm:ss).
- `%w` - Dia da semana (0-6).

Outros formatos podem ser obtidos através da página de manual do `date`.
Para maiores detalhes, veja a página de manual do comando `date`.

Para ver a data atual digite: `date`

Se quiser mudar a Data para 25/12 e a hora para 08:15 digite: `date 12250815`

Para mostrar somente a data no formato dia/mês/ano: `date +%d/%m/%Y`

8.3 df

Mostra o espaço livre/ocupado de cada partição.

`df [`*`opções`*`]`

onde:
opções
-a Inclui sistemas de arquivos com 0 blocos.
-h, –human-readable Mostra o espaço livre/ocupado em *MB, KB, GB* ao invés de blocos.
-H Idêntico a `-h` mas usa 1000 ao invés de 1024 como unidade de cálculo.
-k Lista em Kbytes.
-l Somente lista sistema de arquivos locais.
-m Lista em Mbytes (equivalente a –block-size=1048576).

Exemplos: `df`, `df -h`, `df -t vfat`.

8.4 ln

Cria links para arquivos e diretórios no sistema. O link é um mecanismo que faz referência a outro arquivo ou diretório em outra localização. O link em sistemas `GNU/Linux` faz referência reais ao arquivo/diretório podendo ser feita cópia do link (será copiado o arquivo alvo), entrar no diretório (caso o link faça referência a um diretório), etc.

`ln [`*`opções`*`] [`*`origem`*`] [`*`link`*`]`

Onde:
origem Diretório ou arquivo de onde será feito o link.
link Nome do link que será criado.
opções
-s Cria um link simbólico. Usado para criar ligações com o arquivo/diretório de destino.
-v Mostra o nome de cada arquivo antes de fazer o link.
-d Cria um hard link para diretórios. Somente o root pode usar esta opção.
Existem 2 tipos de links: *simbólicos* e *hardlinks*.

- O *link simbólico* cria um arquivo especial no disco (do tipo link) que tem como conteúdo o caminho para chegar até o arquivo alvo (isto pode ser verificado pelo tamanho do arquivo do link). Use a opção `-s` para criar links simbólicos.
- O *hardlink* faz referência ao mesmo inodo do arquivo original, desta forma ele será perfeitamente idêntico, inclusive nas permissões de acesso, ao arquivo original. Ao contrário dos links simbólicos, não é possível fazer um hardlink para um diretório ou fazer referência a arquivos que estejam em partições diferentes.

Observações:
- Se for usado o comando `rm` com um link, somente o link será removido.
- Se for usado o comando `cp` com um link, o arquivo original será copiado ao invés do link.
- Se for usado o comando `mv` com um link, a modificação será feita no link.
- Se for usado um comando de visualização (como o `cat`), o arquivo original será visualizado.

Exemplos:
- `ln -s /dev/ttyS1 /dev/modem` - Cria o link `/dev/modem` para o arquivo `/dev/ttyS1`.
- `ln -s /tmp ~/tmp` - Cria um link `~/tmp` para o diretório `/tmp`.

8.5 du

Mostra o espaço ocupado por arquivos e sub-diretórios do diretório atual.

`du [`*`opções`*`]`

onde:
opções
-a, –all Mostra o espaço ocupado por todos os arquivos.
-b, –bytes Mostra o espaço ocupado em bytes.
-c, –total Faz uma totalização de todo espaço listado.
-D Não conta links simbólicos.
-h, –human Mostra o espaço ocupado em formato legível por humanos (Kb, Mb) ao invés de usar blocos.
-H Como o anterior mas usa 1000 e não 1024 como unidade de cálculo.
-k Mostra o espaço ocupado em Kbytes.
-m Mostra o espaço ocupado em Mbytes.
-S, –separate-dirs Não calcula o espaço ocupado por sub-diretórios.

Exemplo: `du -h`, `du -hc`.

8.6 find

Procura por arquivos/diretórios no disco. `find` pode procurar arquivos através de sua data de modificação, tamanho, etc através do uso de opções. `find`, ao contrário de outros programas,

usa opções longas através de um "-".

`find [diretório] [opções/expressão]`

Onde:
diretório Inicia a procura neste diretório, percorrendo seu sub-diretórios.
opções/expressão
-name [expressão] Procura pelo nome [expressão] nos nomes de arquivos e diretórios processados.
-depth Processa os sub-diretórios primeiro antes de processar os arquivos do diretório principal.
-maxdepth [num] Faz a procura até [num] sub-diretórios dentro do diretório que está sendo pesquisado.
-mindepth [num] Não faz nenhuma procura em diretórios menores que [num] níveis.
-mount, -xdev Não faz a pesquisa em sistemas de arquivos diferentes daquele de onde o comando `find` foi executado.
-size [num] Procura por arquivos que tiverem o tamanho [num]. [num] pode ser antecedido de "+" ou "-" para especificar um arquivo maior ou menor que [num]. A opção -size pode ser seguida de:
- `b` - Especifica o tamanho em blocos de 512 bytes. É o padrão caso [num] não seja acompanhado de nenhuma letra.
- `c` - Especifica o tamanho em bytes.
- `k` - Especifica o tamanho em Kbytes.

-type [tipo] Procura por arquivos do [tipo] especificado. Os seguintes tipos são aceitos:
- `b` - bloco
- `c` - caracter
- `d` - diretório
- `p` - pipe
- `f` - arquivo regular
- `l` - link simbólico
- `s` - sockete

A maior parte dos argumentos numéricos podem ser precedidos por "+" ou "-". Para detalhes sobre outras opções e argumentos, consulte a página de manual.

Exemplo:
- `find / -name grep` - Procura no diretório raíz e sub-diretórios um arquivo/diretório chamado `grep`.
- `find / -name grep -maxdepth 3` - Procura no diretório raíz e sub-diretórios até o 3o. nível, um arquivo/diretório chamado `grep`.
- `find . -size +1000k` - Procura no diretório atual e sub-diretórios um arquivo com tamanho maior que 1000 kbytes (1Mbyte).

8.7 free

Mostra detalhes sobre a utilização da memória RAM do sistema.

`free [opções]`

Onde:
opções
-b Mostra o resultado em bytes.
-k Mostra o resultado em Kbytes.
-m Mostra o resultado em Mbytes.
-o Oculta a linha de buffers.
-t Mostra uma linha contendo o total.
-s [num] Mostra a utilização da memória a cada [num] segundos.
O `free` é uma interface ao arquivo `/proc/meminfo`.

8.8 grep

Procura por um texto dentro de um arquivo(s) ou no dispositivo de entrada padrão.

`grep [`*`expressão`*`] [`*`arquivo`*`] [`*`opções`*`]`

Onde:
expressão palavra ou frase que será procurada no texto. Se tiver mais de 2 palavras você deve identifica-la com aspas “” caso contrário o `grep` assumirá que a segunda palavra é o arquivo!
arquivo Arquivo onde será feita a procura.
opções
-A [número] Mostra o [número] de linhas após a linha encontrada pelo `grep`.
-B [número] Mostra o [número] de linhas antes da linha encontrada pelo `grep`.
-f [arquivo] Especifica que o texto que será localizado, esta no arquivo [arquivo].
-h, –no-filename Não mostra os nomes dos arquivos durante a procura.
-i, –ignore-case Ignora diferença entre maiúsculas e minúsculas no texto procurado e arquivo.
-n, –line-number Mostra o nome de cada linha encontrada pelo `grep`.
-E Ativa o uso de expressões regulares.
-U, –binary Trata o arquivo que será procurado como binário.
Se não for especificado o nome de um arquivo ou se for usado um hífen “-”, `grep` procurará a string no dispositivo de entrada padrão. O `grep` faz sua pesquisa em arquivos texto. Use o comando `zgrep` para pesquisar diretamente em arquivos compactados com `gzip`, os comandos e opções são as mesmas.

Exemplos: `grep "capitulo" texto.txt`, `ps ax|grep inetd`, `grep "capitulo" texto.txt -A 2 -B 2`.

8.9 head

Mostra as linhas iniciais de um arquivo texto.

`head [`*`opções`*`]`

Onde:

-c [numero] Mostra o [numero] de bytes do inicio do arquivo.
-n [numero] Mostra o [numero] de linhas do inicio do arquivo. Caso não for especificado, o `head` mostra as 10 primeiras linhas.

Exemplos: `head teste.txt`, `head -n 20 teste.txt`.

8.10 nl

Mostra o número de linhas junto com o conteúdo de um arquivo.

`nl [`*`opções`*`] [`*`arquivo`*`]`

Onde:
opções
-f [opc] Faz a filtragem de saída de acordo com [opc]:
  a Numera todas as linhas.
  t Não numera linhas vazias.
  n Numera linhas vazias.
  texto Numera somente linhas que contém o [texto].
-v [num] Número inicial (o padrão é 1).
-i [num] Número de linhas adicionadas a cada linha do arquivo (o padrão é 1).

Exemplos: `nl /etc/passwd`, `nl -i 2 /etc/passwd`.

8.11 more

Permite fazer a paginação de arquivos ou da entrada padrão. O comando `more` pode ser usado como comando para leitura de arquivos que ocupem mais de uma tela. Quando toda a tela é ocupada, o `more` efetua uma pausa e permite que você pressione `Enter` ou `espaço` para continuar avançando no arquivo sendo visualizado. Para sair do `more` pressione `q`.

`more [`*`arquivo`*`]`

Onde: *arquivo* É o arquivo que será paginado.

Para visualizar diretamente arquivos texto compactados pelo `gzip .gz` use o comando `zmore`.

Exemplos: `more /etc/passwd`, `cat /etc/passwd|more`.

8.12 less

Permite fazer a paginação de arquivos ou da entrada padrão. O comando `less` pode ser usado como comando para leitura de arquivos que ocupem mais de uma tela. Quando toda a tela é ocupada, o `less` efetua uma pausa (semelhante ao `more`) e permite que você pressione Seta para Cima e Seta para Baixo ou PgUP/PgDown para fazer o rolamento da página. Para sair do `less` pressione `q`.

`less [arquivo]`

Onde: *arquivo* É o arquivo que será paginado.

Para visualizar diretamente arquivos texto compactados pelo utilitário `gzip` (arquivos `.gz`), use o comando `zless`.

Exemplos: `less /etc/passwd`, `cat /etc/passwd|less`

8.13 sort

Organiza as linhas de um arquivo texto ou da entrada padrão.

`sort [opções] [arquivo]`

Onde:
arquivo É o nome do arquivo que será organizado. Caso não for especificado, será usado o dispositivo de entrada padrão (normalmente o teclado ou um "`|`").
opções
-b Ignora linhas em branco.
-d Somente usa letras, dígitos e espaços durante a organização.
-f Ignora a diferença entre maiúsculas e minúsculas.
-r Inverte o resultado da comparação.
-n Caso estiver organizando um campo que contém números, os números serão organizados na ordem aritmética. Por exemplo, se você tiver um arquivo com os números

```
100
10
50
```

Usando a opção `-n`, o arquivo será organizado desta maneira:

```
10
50
100
```

Caso esta opção **não** for usada com o `sort`, ele organizará como uma listagem alfabética (que começam de `a` até `z` e do `0` até `9`)

```
10
100
50
```

-c Verifica se o arquivo já esta organizado. Caso não estiver, retorna a mensagem "disorder on *arquivo*".
-o *arquivo* Grava a saída do comando `sort` no *arquivo*.

Abaixo, exemplos de uso do comando `sort`:

- `sort texto.txt` - Organiza o arquivo `texto.txt` em ordem crescente.
- `sort texto.txt -r` - Organiza o conteúdo do arquivo `texto.txt` em ordem decrescente.
- `cat texto.txt|sort` - Faz a mesma coisa que o primeiro exemplo, só que neste caso a saída do comando `cat` é redirecionado a entrada padrão do comando `sort`.

- `sort -f texto.txt` - Ignora diferenças entre letras maiúsculas e minúsculas durante a organização.

8.14 tail

Mostra as linhas finais de um arquivo texto.

`tail [opções]`

Onde:
-c [numero] Mostra o [numero] de bytes do final do arquivo.
-n [numero] Mostra o [numero] de linhas do final do arquivo.
-f Mostra continuamente linhas adicionadas no final do arquivo.

Exemplos: `tail teste.txt`,`tail -n 20 teste.txt`.

8.15 time

Mede o tempo gasto para executar um processo (programa).

`time [comando]`

Onde: *comando* é o comando/programa que deseja medir o tempo gasto para ser concluído.

Exemplo: `time ls`,`time find / -name crontab`.

8.16 touch

Muda a data e hora que um arquivo foi criado. Também pode ser usado para criar arquivos vazios. Caso o `touch` seja usado com arquivos que não existam, por padrão ele criará estes arquivos.

`touch [opções] [arquivos]`

Onde:
arquivos Arquivos que terão sua data/hora modificados.
opções
-t MMDDhhmm[ANO.segundos] Usa Mês (MM), Dias (DD), Horas (hh), minutos (mm) e opcionalmente o ANO e segundos para modificação do(s) arquivos ao invés da data e hora atual.
-a, –time=atime Faz o `touch` mudar somente a data e hora do acesso ao arquivo.
-c, –no-create Não cria arquivos vazios, caso os *arquivos* não existam.
-m, –time=mtime Faz o `touch` mudar somente a data e hora da modificação.
-r [arquivo] Usa as horas no [arquivo] como referência ao invés da hora atual.

Exemplos:

- `touch teste` - Cria o arquivo `teste` caso ele não existir.
- `touch -t 10011230 teste` - Altera da data e hora do arquivo para 01/10 e 12:30.
- `touch -t 120112301999.30 teste` - Altera da data, hora ano, e segundos do arquivo para 01/12/1999 e 12:30:30.
- `touch -t 12011200 *` - Altera a data e hora do arquivo para 01/12 e 12:00.

8.17 uptime

Mostra o tempo de execução do sistema desde que o computador foi ligado.

`uptime`

8.18 dmesg

Mostra as mensagens de inicialização do kernel. São mostradas as mensagens da última inicialização do sistema.

`dmesg | less`

8.19 mesg

Permite ou não o recebimentos de requisições de `talk` de outros usuários.

`mesg [y/n]`

Onde: *y* permite que você receba "talks" de outros usuários.

Digite `mesg` para saber se você pode ou não receber "talks" de outros usuários. Caso a resposta seja "n" você poderá enviar um talk para alguém mas o seu sistema se recusará em receber talks de outras pessoas.

É interessante colocar o comando `mesg y` em seu arquivo de inicialização `.bash_profile` para permitir o recebimento de "talks" toda vez que entrar no sistema.

Para detalhes sobre como se comunicar com outros usuários, veja o comando 'talk' on page 85.

8.20 echo

Mostra mensagens. Este comando é útil na construção de scripts para mostrar mensagens na tela para o usuário acompanhar sua execução.

`echo [mensagem]`

A opção `-n` pode ser usada para que não ocorra o salto de linha após a mensagem ser mostrada.

8.21 su

Permite o usuário mudar sua identidade para outro usuário sem fazer o logout. Útil para executar um programa ou comando como root sem ter que abandonar a seção atual.

`su [usuário] [-c comando]`

Onde: *usuário* é o nome do usuário que deseja usar para acessar o sistema. Se não digitado, é assumido o usuário `root`. Caso seja especificado *-c comando*, executa o comando sob o usuário especificado.

Será pedida a senha do superusuário para autenticação. Digite `exit` quando desejar retornar a identificação de usuário anterior.

8.22 sync

Grava os dados do cache de disco na memória RAM para todos os discos rígidos e flexíveis do sistema. O cache um mecanismo de aceleração que permite que um arquivo seja armazenado na memória ao invés de ser imediatamente gravado no disco, quando o sistema estiver ocioso, o arquivo é gravado para o disco. O `GNU/Linux` procura utilizar toda memória RAM disponível para o cache de programas acelerando seu desempenho de leitura/gravação.

`sync`

O uso do `sync` é útil em disquetes quando gravamos um programa e precisamos que os dados sejam gravados imediatamente para retirar o disquete da unidade. Mas o método recomendado é especificar a opção `sync` durante a montagem da unidade de disquetes (para detalhes veja 'fstab' on page 49.

8.23 uname

Retorna o nome e versão do kernel atual.

`uname`

8.24 reboot

Reinicia o computador.

8.25 shutdown

Desliga/reinicia o computador imediatamente ou após determinado tempo (programável) de forma segura. Todos os usuários do sistema são avisados que o computador será desligado

. Este comando somente pode ser executado pelo usuário root ou quando é usada a opção `-a` pelos usuários cadastrados no arquivo `/etc/shutdown.allow` que estejam logados no console virtual do sistema.

`shutdown [`*opções*`] [`*hora*`] [`*mensagem*`]`

hora Momento que o computador será desligado. Você pode usar `HH:MM` para definir a hora e minuto, `MM` para definir minutos, `+SS` para definir após quantos segundos, ou `now` para imediatamente (equivalente a +0). O `shutdown` criará o arquivo `/etc/nologin` para não permitir que novos usuários façam login no sistema (com excessão do root). Este arquivo é removido caso a execução do `shutdown` seja cancelada (opção -c) ou após o sistema ser reiniciado.

mensagem Mensagem que será mostrada a todos os usuários alertando sobre o reinicio/desligamento do sistema.

opções

-h Inicia o processo para desligamento do computador.

-r Reinicia o sistema

-c Cancela a execução do `shutdown`. Você pode acrescentar uma mensagem avisando aos usuários sobre o fato.

O `shutdown` envia uma mensagem a todos os usuários do sistema alertando sobre o desligamento durante os 15 minutos restantes e assim permite que finalizem suas tarefas. Após isto, o `shutdown` muda o nível de execução através do comando `init` para 0 (desligamento), 1 (modo monousuário), 6 (reinicialização). É recomendado utilizar o símbolo "&" no final da linha de comando para que o `shutdown` seja executado em segundo plano.

Quando restarem apenas 5 minutos para o reinicio/desligamento do sistema, o programa `login` será desativado, impedindo a entrada de novos usuários no sistema.

O programa `shutdown` pode ser chamado pelo `init` através do pressionamento da combinação das teclas de reinicialização `CTRL+ALT+DEL` alterando-se o arquivo `/etc/inittab`. Isto permite que somente os usuários autorizados (ou o root) possam reinicializar o sistema.

Exemplos:

- "`shutdown -h now`" - Desligar o computador imediatamente.
- "`shutdown -r now`" - Reinicia o computador imediatamente.
- "`shutdown 19:00 A manutenção do servidor será iniciada às 19:00`" - Faz o computador entrar em modo monousuário (init 1) às 19:00 enviando a mensagem *A manutenção do servidor será iniciada às 19:00* a todos os usuários conectados ao sistema.
- "`shutdown -r 15:00 O sistema será reiniciado às 15:00 horas`" - Faz o computador ser reiniciado (init 6) às 15:00 horas enviando a mensagem *O sistema será reiniciado às 15:00 horas* a todos os usuários conectados ao sistema.
- `shutdown -r 20` - Faz o sistema ser reiniciado após 20 minutos.
- `shutdown -c` - Cancela a execução do `shutdown`.

8.26 wc

Conta o número de palavras, bytes e linhas em um arquivo ou entrada padrão. Se as opções forem omitidas, o `wc` mostra a quantidade de linhas, palavras, e bytes.

`wc [opções] [arquivo]`

Onde:
arquivo Arquivo que será verificado pelo comando `wc`.
opções
-c, –bytes Mostra os bytes do arquivo.
-w, –words Mostra a quantidade de palavras do arquivo.
-l, –lines Mostra a quantidade de linhas do arquivo.
A ordem da listagem dos parâmetros é única, e modificando a posição das opções não modifica a ordem que os parâmetros são listados.

Exemplo:
- `wc /etc/passwd` - Mostra a quantidade de linhas, palavras e letras (bytes) no arquivo `/etc/passwd`.
- `wc -w /etc/passwd` - Mostra a quantidade de palavras.
- `wc -l /etc/passwd` - Mostra a quantidade de linhas.
- `wc -l -w /etc/passwd` - Mostra a quantidade de linhas e palavras no arquivo `/etc/passwd`.

8.27 seq

Imprime uma seqüência de números começando em [primeiro] e terminando em [último], utilizando [incremento] para avançar.

`seq [opções] [primeiro] [incremento] [último]`

Onde:
primeiro Número inicial da seqüência.
incremento Número utilizado para avançar na seqüência.
último Número final da seqüência.
opções
-f, –format=[formato] Formato de saída dos números da seqüência. Utilize o estilo do printf para ponto flutuante (valor padrão: %g).
-s, –separator=[string] Usa [string] para separar a seqüência de números (valor padrão: \n).
-w, –equal-width Insere zeros na frente dos números mantendo a seqüência alinhada.

Observações:
- Se [primeiro] ou [incremento] forem omitidos, o valor padrão 1 será utilizado.
- Os números recebidos são interpretados como números em ponto flutuante.

incremento deve ser positivo se [primeiro] for menor do que o último, e negativo caso contrário.
- Quando utilizarmos a opção –format, o argumento deve ser exatamente %e, %f ou %g.

Exemplos: `seq 0 2 10`, `seq -w 0 10`, `seq -f%f 0 10`, `seq -s", " 0 10`

Capítulo 9

Comandos de rede

Este capítulo traz alguns comandos úteis para uso em rede e ambientes multiusuário.

9.1 who

Mostra quem está atualmente conectado no computador. Este comando lista os nomes de usuários que estão conectados em seu computador, o terminal e data da conexão.

`who [opções]`

onde:
opções
-H, –heading Mostra o cabeçalho das colunas.
-b, –boot Mostra o horário do último boot do sistema.
-d, –dead Mostra processos mortos no sistema.
-i, -u, –idle Mostra o tempo que o usuário está parado em Horas:Minutos.
-m, i am Mostra o nome do computador e usuário associado ao nome. É equivalente a digitar `who i am` ou `who am i`.
-q, –count Mostra o total de usuários conectados aos terminais.
-r, –runlevel Mostra o nível de execução atual do sistema e desde quando ele está ativo.
-T, -w, –mesg Mostra se o usuário pode receber mensagens via `talk` (conversação).

- + O usuário recebe mensagens via talk
- - O usuário não recebe mensagens via talk.
- ? Não foi possível determinar o dispositivo de terminal onde o usuário está conectado.

9.2 telnet

Permite acesso a um computador remoto. É mostrada uma tela de acesso correspondente ao computador local onde deve ser feita a autenticação do usuário para entrar no sistema.

Muito útil, mas deve ser tomado cuidados ao disponibilizar este serviço para evitar riscos de segurança e usado o `ssh` sempre que possível por ser um protocolo criptografado e com recursos avançados de segurança.

`telnet [`*opções*`] [`*ip/dns*`] [`*porta*`]`

onde:
ip/dns Endereço IP do computador de destino ou nome DNS.
porta Porta onde será feita a conexão. Por padrão, a conexão é feita na porta *23*.
opções **-8** Requisita uma operação binária de 8 bits. Isto força a operação em modo binário para envio e recebimento. Por padrão, `telnet` não usa 8 bits.
-a Tenta um login automático, enviando o nome do usuário lido da variável de ambiente `USER`.
-d Ativa o modo de debug.
-r Ativa a emulação de rlogin.
-l [usuário] Faz a conexão usando [usuário] como nome de usuário.

Exemplo: `telnet 192.168.1.1`,`telnet 192.168.1.1 23`.

9.3 finger

Mostra detalhes sobre os usuários de um sistema. Algumas versões do `finger` possuem bugs e podem significar um risco para a segurança do sistema. É recomendado desativar este serviço na máquina local.

`finger [`*usuário*`] [`*usuário@host*`]`

Onde:
usuário Nome do usuário que deseja obter detalhes do sistema. Se não for digitado o nome de usuário, o sistema mostra detalhes de todos os usuários conectados no momento.
usuário@host Nome do usuário e endereço do computador que deseja obter detalhes.
-l Mostra os detalhes de todos os usuários conectados no momento. Entre os detalhes, estão incluídos o *nome do interpretador de comandos* (shell) do usuário, *diretório home*, *nome do usuário*, *endereço*, etc.
-p Não exibe o conteúdo dos arquivos `.plan` e `.project`
Se for usado sem parâmetros, mostra os dados de todos os usuários conectados atualmente ao seu sistema.

Exemplo: `finger`, `finger root`.

9.4 ftp

Permite a transferência de arquivos do computador remoto/local e vice versa. O file transfer protocol é o sistema de transmissão de arquivos mais usado na Internet. É requerida a autenticação do usuário para que seja permitida a conexão. Muitos servidores ftp disponibilizam acesso anônimo aos usuários, com acesso restrito.

Uma vez conectado a um servidor `ftp`, você pode usar a maioria dos comandos do `GNU/Linux` para operá-lo.

`ftp [`*`ip/dns`*`]`

Abaixo alguns dos comandos mais usados no FTP:
ls Lista arquivos do diretório atual.
cd [diretório] Entra em um diretório.
get [arquivo] Copia um arquivo do servidor ftp para o computador local. O arquivo é gravado, por padrão, no diretório onde o programa ftp foi executado.
hash [on/off] Por padrão esta opção está desligada. Quando ligada, faz com que o caracter "#" seja impresso na tela indicando o progresso do download.
mget [arquivos] Semelhante ao get, mas pode copiar diversos arquivos e permite o uso de curingas.
send [arquivo] Envia um arquivo para o diretório atual do servidor FTP (você precisa de uma conta com acesso a gravação para fazer isto).
prompt [on/off] Ativa ou desativa a pergunta para a cópia de arquivo. Se estiver como `off` assume sim para qualquer pergunta.

Exemplo: `ftp ftp.debian.org`.

9.5 whoami

Mostra o nome que usou para se conectar ao sistema. É útil quando você usa várias contas e não sabe com qual nome entrou no sistema :-)

`whoami`

9.6 dnsdomainname

Mostra o nome do domínio de seu sistema.

9.7 hostname

Mostra ou muda o nome de seu computador na rede.

9.8 talk

Inicia conversa com outro usuário de sistema em uma rede local ou Internet. Talk é um programa de conversação em tempo real onde uma pessoa vê o que a outra escreve.

`talk [`*`usuário`*`] [`*`tty`*`]`

ou

`talk [`*`usuário@host`*`]`

Onde:

usuário Nome de login do usuário que deseja iniciar a conversação. Este nome pode ser obtido com o comando `who` (veja 'who' on page 83).

tty O nome de terminal onde o usuário está conectado, para iniciar uma conexão local.

usuário@host Se o usuário que deseja conversar estiver conectado em um computador remoto, você deve usar o nome do usuário@hosname do computador.

Após o `talk` ser iniciado, ele verificará se o usuário pode receber mensagens, em caso positivo, ele enviará uma mensagem ao usuário dizendo como responder ao seu pedido de conversa. Veja 'who' on page 83.

Para poder fazer a rolagem para cima e para baixo no `talk`, pressione `CTRL+P`(Previous - Tela anterior) e `CTRL+N` (Next - Próxima tela).

Você deve autorizar o recebimento de talks de outros usuários para que eles possam se comunicar com você , para detalhes veja o comando 'mesg' on page 77.

Capítulo 10

Comandos para manipulação de contas

Este capítulo traz comandos usados para manipulação de conta de usuários e grupos em sistemas GNU/Linux. Entre os assuntos descritos aqui estão adicionar usuários ao sistema, adicionar grupos, incluir usuários em grupos existentes, etc.

10.1 adduser

Adiciona um usuário ou grupo no sistema. Por padrão, quando um novo usuário é adicionado, é criado um grupo com o mesmo nome do usuário. Opcionalmente o `adduser` também pode ser usado para adicionar um usuário a um grupo (veja 'Adicionando o usuário a um grupo extra' on page 90). Será criado um diretório home com o nome do usuário (a não ser que o novo usuário criado seja um usuário do sistema) e este receberá uma identificação. A identificação do usuário (UID) escolhida será a primeira disponível no sistema especificada de acordo com a faixa de UIDS de usuários permitidas no arquivo de configuração `/etc/adduser.conf`. Este é o arquivo que contém os padrões para a criação de novos usuários no sistema.

`adduser [`*`opções`*`] [`*`usuário/grupo`*`]`

Onde:
usuário/grupo Nome do novo usuário que será adicionado ao sistema.
opções

-disable-passwd Não executa o programa `passwd` para escolher a senha e somente permite o uso da conta após o usuário escolher uma senha.

–force-badname Desativa a checagem de senhas ruins durante a adição do novo usuário. Por padrão o `adduser` checa se a senha pode ser facilmente adivinhada.

–group Cria um novo grupo ao invés de um novo usuário. A criação de grupos também pode ser feita pelo comando `addgroup`.

-uid [num] Cria um novo usuário com a identificação [num] ao invés de procurar o próximo UID disponível.

-gid [num] Faz com que o usuário seja parte do grupo [gid] ao invés de pertencer a um novo grupo que será criado com seu nome. Isto é útil caso deseje permitir que grupos de

usuários possam ter acesso a arquivos comuns. Caso estiver criando um novo grupo com `adduser`, a identificação do novo grupo será [num].

–home [dir] Usa o diretório [dir] para a criação do diretório home do usuário ao invés de usar o especificado no arquivo de configuração `/etc/adduser.conf`.

–ingroup [nome] Quando adicionar um novo usuário no sistema, coloca o usuário no grupo [nome] ao invés de criar um novo grupo.

–quiet Não mostra mensagens durante a operação.

–system Cria um usuário de sistema ao invés de um usuário normal.

Os dados do usuário são colocados no arquivo `/etc/passwd` após sua criação e os dados do grupo são colocados no arquivo `/etc/group`.

OBSERVAÇÃO: Caso esteja usando senhas ocultas (shadow passwords), as senhas dos usuários serão colocadas no arquivo `/etc/shadow` e as senhas dos grupos no arquivo `/etc/gshadow`. Isto aumenta mais a segurança do sistema porque somente o usuário `root` pode ter acesso a estes arquivos, ao contrário do arquivo `/etc/passwd` que possui os dados de usuários e devem ser lidos por todos.

10.2 addgroup

Adiciona um novo grupo de usuários no sistema. As opções usadas são as mesmas do 'adduser' on the preceding page.

`addgroup [`*`usuário/grupo`*`] [`*`opções`*`]`

10.3 passwd

Modifica a parametros e senha de usuário. Um usuário somente pode alterar a senha de sua conta, mas o superusuário (`root`) pode alterar a senha de qualquer conta de usuário, inclusive a data de validade da conta, etc. Os donos de grupos também podem alterar a senha do grupo com este comando.

Os dados da conta do usuário como nome, endereço, telefone, também podem ser alterados com este comando.

`passwd [`*`usuário`*`] [`*`opções`*`]`

Onde:

usuário Nome do usuário que terá sua senha alterada.

opções

-e Força a expiração de senha para a conta especificada.

-k Somente altera a senha se a conta estiver expirada.

Procure sempre combinar letras maiúsculas, minúsculas, e números ao escolher suas senhas. Não é recomendado escolher palavras normais como sua senha pois podem ser vulneráveis a ataques de dicionários cracker. Outra recomendação é utilizar *senhas ocultas* em seu sistema (*shadow password*).

Você deve ser o dono da conta para poder modificar a senhas. O usuário root pode modificar/apagar a senha de qualquer usuário.

Exemplo: `passwd root`.

10.4 gpasswd

Modifica parametros e senha de grupo. Um usuário somente pode alterar a senha de seu grupo, mas o superusuário (`root`) pode alterar a senha de qualquer grupo de usuário, inclusive definir o administrador do grupo.

`gpasswd [`*`opções`*`] [`*`usuario`*`] [`*`grupo`*`]`

Onde:
usuário Nome do usuário/grupo que terá sua senha alterada.
opções
-r *usuario grupo* Remove a senha de grupo.
-R *usuario grupo* Desativa o acesso do grupo usando o comando `newgrp`.
-a *usuario grupo* Adiciona o usuário no grupo especificado.
-d *usuario grupo* Apaga o usuário do gurpo especificado.
Quando o grupo não possui senha, somente quem faz parte do grupo pode utilizar o comando new-grp.

Você deve ser o dono da conta para poder modificar a senhas. O usuário root pode modificar/apagar a senha de qualquer usuário.

Exemplo: `gpasswd grupo`, `gpasswd -a gleydson grupo`.

10.5 newgrp

Altera a identificação de grupo do usuário. Para retornar a identificação anterior, digite `exit` e tecle `Enter`. Para executar um comando com outra identificação de grupo de usuário, use o comando 'sg' on the next page.

`newgrp - [`*`grupo`*`]`

Onde:
- Se usado, inicia um novo ambiente após o uso do comando `newgrp` (semelhante a um novo login no sistema), caso contrário, o ambiente atual do usuário é mantido.
grupo Nome do grupo ou número do grupo que será incluído.
Quando este comando é usado, é pedida a senha do grupo que deseja acessar. Caso a senha do grupo esteja incorreta ou não exista senha definida, a execução do comando é negada. A listagem dos grupos que pertence atualmente pode ser feita usando o comando 'id' on page 92.

10.6 userdel

Apaga um usuário do sistema. Quando é usado, este comando apaga todos os dados da conta especificado dos arquivos de contas do sistema.

`userdel [-r] [usuário]`

Onde:
-r Apaga também o diretório HOME do usuário.

OBS: Note que uma conta de usuário não poderá ser removida caso ele estiver no sistema, pois os programas podem precisar ter acesso aos dados dele (como UID, GID) no `/etc/passwd`.

10.7 groupdel

Apaga um grupo do sistema. Quando é usado, este comando apaga todos os dados do grupo especificado dos arquivos de contas do sistema.

`groupdel [grupo]`

Tenha certeza que não existem arquivos/diretórios criados com o grupo apagado através do comando `find`.

OBS: Você não pode remover o grupo primário de um usuário. Remova o usuário primeiro.

10.8 sg

Executa um comando com outra identificação de grupo. A identificação do grupo de usuário é modificada somente durante a execução do comando. Para alterar a identificação de grupo durante sua seção shell, use o comando 'newgrp' on the previous page.

`sg [-] [grupo] [comando]`

Onde:
– Se usado, inicia um novo ambiente durante o uso do comando (semelhante a um novo login e execução do comando), caso contrário, o ambiente atual do usuário é mantido.
grupo Nome do grupo que o comando será executado.
comando Comando que será executado. O comando será executado pelo bash.
Quando este comando é usado, é pedida a senha do grupo que deseja acessar. Caso a senha do grupo esteja incorreta ou não exista senha definida, a execução do comando é negada.

Exemplo: `sg root ls /root`

10.9 Adicionando o usuário a um grupo extra

Para adicionar um usuário em um novo grupo e assim permitir que ele acesse os arquivos/diretórios que pertencem àquele grupo, você deve estar como root e editar o arquivo

/etc/group com o comando vigr. Este arquivo possui o seguinte formato:

NomedoGrupo:senha:GID:usuários

Onde:

NomedoGrupo É o nome daquele grupo de usuários.

senha Senha para ter acesso ao grupo. Caso esteja utilizando senhas ocultas para grupos, as senhas estarão em /etc/gshadow.

GID Identificação numérica do grupo de usuário.

usuarios Lista de usuários que também fazem parte daquele grupo. Caso exista mais de um nome de usuário, eles devem estar separados por vírgula.

Deste modo para acrescentar o usuário "joao" ao grupo audio para ter acesso aos dispositivos de som do Linux, acrescente o nome no final da linha: "audio:x:100:joao". Pronto, basta digitar logout e entrar novamente com seu nome e senha, você estará fazendo parte do grupo audio (confira digitando groups ou id).

Outros nomes de usuários podem ser acrescentados ao grupo audio bastando separar os nomes com vírgula. Você também pode usar o comando adduser da seguinte forma para adicionar automaticamente um usuário a um grupo:

```
adduser joao audio
```

Isto adicionaria o usuário "joao" ao grupo audio da mesma forma que fazendo-se a edição manualmente.

10.10 chfn

Muda os dados usados pelo comando 'finger' on page 84.

chfn [*usuário*] [*opções*]

Onde:
usuário Nome do usuário.
opções
-f [nome] Adiciona/altera o nome completo do usuário.
-r [nome] Adiciona/altera o número da sala do usuário.
-w [tel] Adiciona/altera o telefone de trabalho do usuário.
-h [tel] Adiciona/altera o telefone residencial do usuário.
-o [outros] Adiciona/altera outros dados do usuário.
Caso o nome que acompanha as opções (como o nome completo) contenha espaços, use "" para identifica-lo.

Exemplo: chfn -f "Nome do Usuário root" root

10.11 id

Mostra a identificação atual do usuário, grupo primário e outros grupos que pertence.

`id [opções] [usuário]`

Onde:
usuário É o usuário que desejamos ver a identificação, grupos primários e complementares.
opções
-g, –group Mostra somente a identificação do grupo primário.
-G, –groups Mostra a identificação de outros grupos que pertence.
-n, –name Mostra o nome do usuário e grupo ao invés da identificação numérica.
-u, –user Mostra somente a identificação do usuário (user ID).
-r, –real Mostra a identificação real de usuário e grupo, ao invés da efetiva. Esta opção deve ser usada junto com uma das opções: -u, -g, ou -G.
Caso não sejam especificadas opções, `id` mostrará todos os dados do usuário.

Exemplo: `id`, `id --user`, `id -r -u`.

10.12 logname

Mostra seu login (username).

`logname`

10.13 users

Mostra os nomes de usuários usando atualmente o sistema. Os nomes de usuários são mostrados através de espaços sem detalhes adicionais, para ver maiores detalhes sobre os usuários, veja os comandos 'id' on the current page e 'who' on page 83.

`users`

Os nomes de usuários atualmente conectados ao sistema são obtidos do arquivo `/var/log/wtmp`.

10.14 groups

Mostra os grupos que o usuário pertence.

`groups [usuário]`

Exemplo: `groups`, `groups root`

Capítulo 11

Permissões de acesso a arquivos e diretórios

As permissões de acesso protegem o sistema de arquivos Linux do acesso indevido de pessoas ou programas não autorizados.

A permissão de acesso do GNU/Linux também impede que um programa mal intencionado, por exemplo, apague um arquivo que não deve, envie arquivos especiais para outra pessoa ou forneça acesso da rede para que outros usuários invadam o sistema. O sistema GNU/Linux é muito seguro e como qualquer outro sistema seguro e confiável impede que usuários mal intencionados (ou iniciantes que foram enganados) instalem programas enviados por terceiros sem saber para que eles realmente servem e causem danos irreversíveis em seus arquivos, seu micro ou sua empresa.

Esta seção do guia, de inicio, pode ser um pouco dificil de se entender, então recomendo ler e ao mesmo tempo prática-la para uma ótima compreensão. Não se preocupe, também coloquei exemplos para ajuda-lo a entender o sistema de permissões de acesso do ambiente GNU/Linux.

11.1 Donos, Grupos e outros usuários

A idéia básica da segurança no sistema GNU/Linux é definir o acesso aos arquivos por donos, grupos e outros usuários:

dono É a pessoa que criou o arquivo ou o diretório. O nome do dono do arquivo/diretório é o mesmo do usuário usado para entrar no sistema GNU/Linux. Somente o dono pode modificar as permissões de acesso do arquivo. As permissões de acesso do dono de um arquivo somente se aplicam ao dono do arquivo/diretório. A identificação do dono também é chamada de `user id` (UID). A identificação de usuário ao qual o arquivo pertence é armazenada no arquivo `/etc/passwd` e do grupo no arquivo `/etc/group`. Estes são arquivos textos comuns e podem ser editados em qualquer editor de texto, mas utilize preferencialmente os comandos `vipw` e `vigr` que executa procedimentos adicionais de checagem de uids e grupos após a alteração. Tenha cuidado para não modificar o campo

que contém a senha do usuário encriptada (que pode estar armazenada no arquivo /etc/passwd caso não estiver usando senhas ocultas).

grupo Permite que vários usuários diferentes tenham acesso a um mesmo arquivo (já que somente o dono poderia ter acesso ao arquivo). Cada usuário pode fazer parte de um ou mais grupos e então acessar arquivos que pertençam ao mesmo grupo que o seu (mesmo que estes arquivos tenham outro *dono*). Por padrão, quando um novo usuário é criado e não especificar nenhum grupo, ele pertencerá ao grupo de mesmo nome do seu grupo primário (este comportamento é controlado pelo parametro `USERGROUPS=yes` do arquivo `/etc/adduser.conf`, veja 'id' on page 92). A identificação do grupo é chamada de `GID (group id)`. Um usuário pode pertencer a um ou mais grupos. Para detalhes de como incluir o usuário em mais grupos veja 'Adicionando o usuário a um grupo extra' on page 90.

outros É a categoria de usuários que não são donos ou não pertencem ao grupo do arquivo.

Cada um dos tipos acima possuem três tipos básicos de permissões de acesso que serão vistas na próxima seção.

11.2 Tipos de Permissões de Acesso

Quanto aos tipos de permissões que se aplicam ao *dono*, *grupo* e *outros usuários*, temos 3 permissões básicas:

- `r` - Permissão de leitura para arquivos. Caso for um diretório, permite listar seu conteúdo (através do comando `ls`, por exemplo).
- `w` - Permissão de gravação para arquivos. Caso for um diretório, permite a gravação de arquivos ou outros diretórios dentro dele. Para que um arquivo/diretório possa ser apagado, é necessário o acesso a gravação.
- `x` - Permite executar um arquivo (caso seja um programa executável). Caso seja um diretório, permite que seja acessado através do comando `cd` (veja 'cd' on page 62 para detalhes).

As permissões de acesso a um arquivo/diretório podem ser visualizadas com o uso do comando `ls -la`. Para maiores detalhes veja 'ls' on page 61. As 3 letras (rwx) são agrupadas da seguinte forma:

```
-rwxr-xr--   gleydson   users  teste
```

Virou uma bagunça não? Vou explicar cada parte para entender o que quer dizer as 10 letras acima (da esquerda para a direita):

- A primeira letra diz qual é o tipo do arquivo. Caso tiver um "d" é um diretório, um "l" um link a um arquivo no sistema (veja 'ln' on page 70 para detalhes) , um "-" quer dizer que é um arquivo comum, etc.

- Da segunda a quarta letra (rwx) dizem qual é a permissão de acesso ao *dono* do arquivo. Neste caso *gleydson* ele tem a permissão de ler (r - read), gravar (w - write) e executar (x - execute) o arquivo `teste`.

- Da quinta a sétima letra (r-x) diz qual é a permissão de acesso ao *grupo* do arquivo. Neste caso todos os usuários que pertencem ao grupo *users* tem a permissão de ler (r), e também executar (x) o arquivo `teste`.

- Da oitava a décima letra (r–) diz qual é a permissão de acesso para os *outros usuários*. Neste caso todos os usuários que não são donos do arquivo `teste` tem a permissão somente para ler o programa.

Veja o comando 'chmod' on page 99 para detalhes sobre a mudança das permissões de acesso de arquivos/diretórios.

11.3 Etapas para acesso a um arquivo/diretório

O acesso a um arquivo/diretório é feito verificando primeiro se o usuário que acessará o arquivo é o seu *dono*, caso seja, as permissões de dono do arquivo são aplicadas. Caso não seja o *dono* do arquivo/diretório, é verificado se ele pertence ao grupo correspondente, caso pertença, as permissões do *grupo* são aplicadas. Caso não pertença ao *grupo*, são verificadas as permissões de acesso para os outros usuários que não são *donos* e não pertencem ao *grupo* correspondente ao arquivo/diretório.

Após verificar aonde o usuário se encaixa nas permissões de acesso do arquivo (se ele é o *dono*, pertence ao *grupo*, ou *outros usuários*), é verificado se ele terá permissão acesso para o que deseja fazer (ler, gravar ou executar o arquivo), caso não tenha, o acesso é negado, mostrando uma mensagem do tipo: "Permission denied" (permissão negada).

O que isto que dizer é que mesmo que você seja o dono do arquivo e definir o acesso do *dono* (através do comando `chmod`) como somente leitura (r) mas o acesso dos *outros usuários* como leitura e gravação, você somente poderá ler este arquivo mas os outros usuários poderão ler/grava-lo.

As permissões de acesso (leitura, gravação, execução) para donos, grupos e outros usuários são independentes, permitindo assim um nível de acesso diferenciado. Para maiores detalhes veja 'Tipos de Permissões de Acesso' on the preceding page.

Lembre-se: Somente o dono pode modificar as permissões de um arquivo/diretório!

Para mais detalhes veja os comandos 'chown' on page 100 e 'chgrp' on page 100.

11.4 Exemplos práticos de permissões de acesso

Abaixo dois exemplos práticos de permissão de acesso: 'Exemplo de acesso a um arquivo' on the following page e a 'Exemplo de acesso a um diretório' on the next page. Os dois exemplos são explicados passo a passo para uma perfeita compreensão do assunto. Vamos a prática!

11.4.1 Exemplo de acesso a um arquivo

Abaixo um exemplo e explicação das permissões de acesso a um arquivo no `GNU/Linux` (obtido com o comando `ls -la`, explicarei passo a passo cada parte:

-rwxr-xr– 1 gleydson user 8192 nov 4 16:00 teste

`-rwxr-xr--` Estas são as permissões de acesso ao arquivo `teste`. Um conjunto de 10 letras que especificam o tipo do arquivo, permissão do dono do arquivo, grupo do arquivo e outros usuários. Veja a explicação detalhada sobre cada uma abaixo:

- **-rwxr-xr–** A primeira letra (do conjunto das 10 letras) determina o tipo do arquivos. Se a letra for um **d** é um diretório, e você poderá acessa-lo usando o comando `cd`. Caso for um **l** é um link simbólico para algum arquivo ou diretório no sistema (para detalhes veja o comando 'ln' on page 70 . Um **-** significa que é um arquivo normal.
- **-rwxr-xr–** Estas 3 letras (da segunda a quarta do conjunto das 10 letras) são as permissões de acesso do *dono* do arquivo `teste`. O dono (neste caso *gleydson*) tem a permissão para ler (r), gravar (w) e executar (x) o arquivo `teste`.
- **-rwxr-xr–** Estes 3 simbolos (do quinto ao sétimo do conjunto de 10) são as permissões de acesso dos usuários que pertencem ao *grupo user* do arquivo `teste`. Os usuários que pertencem ao grupo *user* tem a permissão somente para ler (r) e executar (x) o arquivo `teste` não podendo modifica-lo ou apaga-lo.
- **-rwxr-xr–** Estes 3 simbolos (do oitavo ao décimo) são as permissões de acesso para usuários que **não** são *donos* do arquivo `teste` e que **não** pertencem ao grupo *user*. Neste caso, estas pessoas somente terão a permissão para ver o conteúdo do arquivo `teste`.

gleydson Nome do dono do arquivo `teste`.

user Nome do grupo que o arquivo `teste` pertence.

`teste` Nome do arquivo.

11.4.2 Exemplo de acesso a um diretório

Abaixo um exemplo com explicações das permissões de acesso a um diretório no `GNU/Linux`:

drwxr-x— 2 gleydson user 1024 nov 4 17:55 exemplo

`drwxr-x---` Permissões de acesso ao diretório `exemplo`. É um conjunto de 10 letras que especificam o tipo de arquivo, permissão do dono do diretório, grupo que o diretório pertence e permissão de acesso a outros usuários. Veja as explicações abaixo:

- **drwxr-x—** A primeira letra (do conjunto das 10) determina o tipo do arquivo. Neste caso é um diretório porque tem a letra **d**.
- **drwxr-x—** Estas 3 letras (da segunda a quarta) são as permissões de acesso do *dono* do diretório `exemplo`. O dono do diretório (neste caso *gleydson*) tem a permissão para listar arquivos do diretório (r), gravar arquivos no diretório (w) e entrar no diretório (x).
- **drwxr-x—** Estas 3 letras (da quinta a sétima) são as permissões de acesso dos usuários que pertencem ao *grupo user*. Os usuários que pertencem ao grupo *user* tem a permissão somente para listar arquivos do diretório (r) e entrar no diretório (x) `exemplo`.

drwxr-x— Estes 3 simbolos (do oitavo ao décimo) são as permissões de acesso para usuários que **não** são *donos* do diretório `exemplo` e que **não** pertencem ao grupo *user*. Com as permissões acima, nenhum usuário que se encaixe nas condições de *dono* e *grupo* do diretório tem a permissão de acessa-lo.

gleydson Nome do dono do diretório `exemplo`.

user Nome do grupo que diretório `exemplo` pertence.

`exemplo` Nome do diretório.

Para detalhes de como alterar o dono/grupo de um arquivo/diretório, veja os comandos 'chmod' on page 99, 'chgrp' on page 100 e 'chown' on page 100.

OBSERVAÇÕES:

- O usuário `root` não tem nenhuma restrição de acesso ao sistema.
- Se você tem permissões de gravação no diretório e tentar apagar um arquivo que você não tem permissão de gravação, o sistema perguntará se você confirma a exclusão do arquivo apesar do modo leitura. Caso você tenha permissões de gravação no arquivo, o arquivo será apagado por padrão sem mostrar nenhuma mensagem de erro (a não ser que seja especificada a opção -i com o comando `rm`).
- Por outro lado, mesmo que você tenha permissões de gravação em um arquivo mas não tenha permissões de gravação em um diretório, a exclusão do arquivo será negada.

Isto mostra que é levado mais em consideração a permissão de acesso do diretório do que as permissões dos arquivos e sub-diretórios que ele contém. Este ponto é muitas vezes ignorado por muitas pessoas e expõem seu sistema a riscos de segurança. Imagine o problema que algum usuário que não tenha permissão de gravação em um arquivo mas que a tenha no diretório pode causar em um sistema mal administrado.

11.5 Permissões de Acesso Especiais

Em adição as três permissões básicas (rwx), existem permissões de acesso especiais (stX) que afetam os arquivos e diretórios:

- `s` - Quando é usado na permissão de acesso do *Dono*, ajusta a identificação efetiva do usuário do processo durante a execução de um programa, também chamado de *bit setuid*. Não tem efeito em diretórios. Quando `s` é usado na permissão de acesso do *Grupo*, ajusta a identificação efetiva do grupo do processo durante a execução de um programa, chamado de *bit setgid*. É identificado pela letra `s` no lugar da permissão de execução do grupo do arquivo/diretório. Em diretórios, força que os arquivos criados dentro dele pertençam ao mesmo grupo do diretório, ao invés do grupo primário que o usuário pertence. Ambos *setgid* e *setuid* podem aparecer ao mesmo tempo no mesmo arquivo/diretório. A permissão de acesso especial `s` somente pode aparecer no campo *Dono* e *Grupo*.
- `S` - Idêntico a "s". Significa que não existe a permissão "x" (execução ou entrar no diretório) naquela posição. Um exemplo é o chmod 2760 em um diretório.
- `t` - Salva a imagem do texto do programa no dispositivo swap, assim ele será carregado mais rapidamente quando executado, também chamado de *stick bit*. Em diretórios, impede que outros usuários removam arquivos dos quais não são donos. Isto é chamado de colocar o diretório em modo `append-only`. Um exemplo de diretório que se en-

caixa perfeitamente nesta condição é o /tmp, todos os usuários devem ter acesso para que seus programas possam criar os arquivos temporários lá, mas nenhum pode apagar arquivos dos outros. A permissão especial t, pode ser especificada somente no campo outros usuários das permissões de acesso.

- T - Idêntico a "t". Significa que não existe a permissão "x" naquela posição (por exemplo, em um chmod 1776 em um diretório).
- X - Se você usar X ao invés de x, a permissão de execução somente é aplicada se o arquivo já tiver permissões de execução. Em diretórios ela tem o mesmo efeito que a permissão de execução x.

- Exemplo da permissão de acesso especial X:
  1. Crie um arquivo `teste` (digitando `touch teste`) e defina sua permissão para `rw-rw-r--` (`chmod ug=rw,o=r teste` ou `chmod 664 teste`).
  2. Agora use o comando `chmod a+X teste`
  3. digite `ls -l`
  4. Veja que as permissões do arquivo não foram afetadas.
  5. agora digite `chmod o+x teste`
  6. digite `ls -l`, você colocou a permissão de execução para os outros usuários.
  7. Agora use novamente o comando `chmod a+X teste`
  8. digite `ls -l`
  9. Veja que agora a permissão de execução foi concedida a todos os usuários, pois foi verificado que o arquivo era executável (tinha permissão de execução para outros usuários).
  10. Agora use o comando `chmod a-X teste`
  11. Ele também funcionará e removerá as permissões de execução de todos os usuários, porque o arquivo `teste` tem permissão de execução (confira digitando `ls -l`).
  12. Agora tente novamente o `chmod a+X teste`
  13. Você deve ter reparado que a permissão de acesso especial X é semelhante a x, mas somente faz efeito quanto o arquivo já tem permissão de execução para o dono, grupo ou outros usuários.

  Em diretórios, a permissão de acesso especial X funciona da mesma forma que x, até mesmo se o diretório não tiver nenhuma permissão de acesso (x).

11.6 A conta root

Esta seção foi retirada do Manual de Instalação da Debian.

A conta root é também chamada de *super usuário*, este é um login que não possui restrições de segurança. A conta root somente deve ser usada para fazer a administração do sistema, e usada o menor tempo possível.

Qualquer senha que criar deverá conter de 6 a 8 caracteres (em sistemas usando crypto) ou até frases inteiras (caso esteja usando MD5, que garante maior segurança), e também poderá conter letras maiúsculas e minúsculas, e também caracteres de pontuação. Tenha um cuidado especial quando escolher sua senha root, porque ela é a conta mais poderosa. Evite palavras de dicionário ou o uso de qualquer outros dados pessoais que podem ser adivinhados.

Se qualquer um lhe pedir senha root, seja extremamente cuidadoso. Você normalmente nunca deve distribuir sua conta root, a não ser que esteja administrando um computador com mais de um administrador do sistema.

Utilize uma conta de usuário normal ao invés da conta root para operar seu sistema. Porque não usar a conta root? Bem, uma razão para evitar usar privilégios root é por causa da facilidade de se cometer danos irreparáveis como root. Outra razão é que você pode ser enganado e rodar um programa *Cavalo de Tróia* – que é um programa que obtém poderes do *super usuário* para comprometer a segurança do seu sistema sem que você saiba.

11.7 chmod

Muda a permissão de acesso a um arquivo ou diretório. Com este comando você pode escolher se usuário ou grupo terá permissões para ler, gravar, executar um arquivo ou arquivos. Sempre que um arquivo é criado, seu dono é o usuário que o criou e seu grupo é o grupo do usuário (exceto para diretórios configurados com a permissão de grupo "s", será visto adiante).

`chmod [opções] [permissões] [diretório/arquivo]`

Onde:
diretório/arquivo Diretório ou arquivo que terá sua permissão mudada.
opções
-v, –verbose Mostra todos os arquivos que estão sendo processados.
-f, –silent Não mostra a maior parte das mensagens de erro.
-c, –change Semelhante a opção -v, mas só mostra os arquivos que tiveram as permissões alteradas.
-R, –recursive Muda permissões de acesso do *diretório/arquivo* no diretório atual e subdiretórios.
ugoa+-=rwxXst
- *ugoa* - Controla que nível de acesso será mudado. Especificam, em ordem, usuário (u), grupo (g), outros (o), todos (a).
- +-= - + coloca a permissão, - retira a permissão do arquivo e = define a permissão exatamente como especificado.
- rwx - *r* permissão de leitura do arquivo. *w* permissão de gravação. *x* permissão de execução (ou acesso a diretórios).

`chmod` não muda permissões de links simbólicos, as permissões devem ser mudadas no arquivo alvo do link. Também podem ser usados códigos numéricos octais para a mudança das permissões de acesso a arquivos/diretórios. Para detalhes veja 'Modo de permissão octal' on page 101.

DICA: É possível copiar permissões de acesso do arquivo/diretório, por exemplo, se o arquivo `teste.txt` tiver a permissão de acesso `r-xr-----` e você digitar `chmod o=u`, as permissões de acesso dos outros usuários (o) serão idênticas ao do dono (u). Então a nova permissão de acesso do arquivo `teste.txt` será `r-xr--r-x`

Exemplos de permissões de acesso:
`chmod g+r *` Permite que todos os usuários que pertençam ao grupo dos arquivos (g) tenham (+) permissões de leitura (r) em todos os arquivos do diretório atual.

chmod o-r teste.txt Retira (-) a permissão de leitura (r) do arquivo `teste.txt` para os outros usuários (usuários que não são donos e não pertencem ao grupo do arquivo `teste.txt`).

chmod uo+x teste.txt Inclui (+) a permissão de execução do arquivo `teste.txt` para o dono e outros usuários do arquivo.

chmod a+x teste.txt Inclui (+) a permissão de execução do arquivo `teste.txt` para o dono, grupo e outros usuários.

chmod a=rw teste.txt Define a permissão de todos os usuários exatamente (=) para leitura e gravação do arquivo `teste.txt`.

11.8 chgrp

Muda o grupo de um arquivo/diretório.

`chgrp [opções] [grupo] [arquivo/diretório]`

Onde:

grupo Novo grupo do *arquivo/diretório*.

arquivo/diretório Arquivo/diretório que terá o grupo alterado.

opções

-c, –changes Somente mostra os arquivos/grupos que forem alterados.

-f, –silent Não mostra mensagens de erro para arquivos/diretórios que não puderam ser alterados.

-v, –verbose Mostra todas as mensagens e arquivos sendo modificados.

-R, –recursive Altera os grupos de arquivos/sub-diretórios do diretório atual.

11.9 chown

Muda dono de um arquivo/diretório. Opcionalmente pode também ser usado para mudar o grupo.

`chown [opções] [dono.grupo] [diretório/arquivo]`

onde:

dono.grupo Nome do *dono.grupo* que será atribuído ao *diretório/arquivo*. O grupo é opcional.

diretório/arquivo Diretório/arquivo que o dono.grupo será modificado.

opções

-v, –verbose Mostra os arquivos enquanto são alterados.

-f, –supress Não mostra mensagens de erro durante a execução do programa.

-c, –changes Mostra somente arquivos que forem alterados.

-R, –recursive Altera dono e grupo de arquivos no diretório atual e sub-diretórios.

O *dono.grupo* pode ser especificado usando o nome de grupo ou o código numérico correspondente ao grupo (GID).

Você deve ter permissões de gravação no diretório/arquivo para alterar seu dono/grupo.

- `chown gleydson teste.txt` - Muda o dono do arquivo `teste.txt` para `gleydson`.
- `chown gleydson.foca teste.txt` - Muda o dono do arquivo `teste.txt` para `gleydson` e seu grupo para `foca`.
- `chown -R gleydson.focalinux *` - Muda o dono/grupo dos arquivos do diretório atual e sub-diretórios para `gleydson/focalinux` (desde que você tenha permissões de gravação no diretórios e sub-diretórios).

11.10 Modo de permissão octal

Ao invés de utilizar os modos de permissão `+r`, `-r`, etc, pode ser usado o modo octal para se alterar a permissão de acesso a um arquivo. O modo octal é um conjunto de oito números onde cada número define um tipo de acesso diferente.

É mais flexível gerenciar permissões de acesso usando o modo octal ao invés do comum, pois você especifica diretamente a permissão do dono, grupo, outros ao invés de gerenciar as permissões de cada um separadamente. Abaixo a lista de permissões de acesso octal:

- `0` - Nenhuma permissão de acesso. Equivalente a -rwx.
- `1` - Permissão de execução (x).
- `2` - Permissão de gravação (w).
- `3` - Permissão de gravação e execução (wx). Equivalente a permissão 2+1
- `4` - Permissão de leitura (r).
- `5` - Permissão de leitura e execução (rx). Equivalente a permissão 4+1
- `6` - Permissão de leitura e gravação (rw). Equivalente a permissão 4+2
- `7` - Permissão de leitura, gravação e execução. Equivalente a +rwx (4+2+1).

O uso de um deste números define a permissão de acesso do *dono*, *grupo* ou *outros usuários*. Um modo fácil de entender como as permissões de acesso octais funcionam, é através da seguinte tabela:

```
1 = Executar
2 = Gravar
4 = Ler

* Para Dono e Grupo, multiplique as permissões acima por x100 e x10.
```

Basta agora fazer o seguinte:

- Somente permissão de execução, use 1.
- Somente a permissão de leitura, use 4.
- Somente permissão de gravação, use 2.
- Permissão de leitura/gravação, use 6 (equivale a 2+4 / Gravar+Ler).
- Permissão de leitura/execução, use 5 (equivale a 1+4 / Executar+Ler).
- Permissão de execução/gravação, use 3 (equivale a 1+2 / Executar+Gravar).
- Permissão de leitura/gravação/execução, use 7 (equivale a 1+2+4 / Executar+Gravar+Ler).

Vamos a prática com alguns exemplos:

```
"chmod 764 teste"
```

Os números são interpretados da **direita para a esquerda** como permissão de acesso aos *outros usuários* (4), *grupo* (6), e *dono* (7). O exemplo acima faz os *outros usuários* (4) terem acesso somente leitura (r) ao arquivo `teste`, o *grupo* (6) ter a permissão de leitura e gravação (w), e o *dono* (7) ter permissão de leitura, gravação e execução (rwx) ao arquivo `teste`.

Outro exemplo:

```
"chmod 40 teste"
```

O exemplo acima define a permissão de acesso dos *outros usuários* (0) como nenhuma, e define a permissão de acesso do *grupo* (4) como somente leitura (r). Note usei somente dois números e então a permissão de acesso do *dono* do arquivo `não` é modificada (leia as permissões de acesso da direita para a esquerda!). Para detalhes veja a lista de permissões de acesso em modo octal no inicio desta seção.

```
"chmod 751 teste"
```

O exemplo acima define a permissão de acesso dos *outros usuários* (1) para somente execução (x), o acesso do *grupo* (5) como leitura e execução (rx) e o acesso do *dono* (7) como leitura, gravação e execução (rwx).

11.11 umask

A umask (*user mask*) são 3 números que definem as permissões iniciais do `dono`, `grupo` e `outros usuários` que o arquivo/diretório receberá quando for criado ou copiado para um novo local. Digite `umask` sem parâmetros para retornar o valor de sua umask atual.

A umask tem efeitos diferentes caso o arquivo que estiver sendo criado for *binário* (um programa executável) ou *texto* ('Arquivo texto e binário' on page 25) . Veja a tabela a seguir para ver qual é a mais adequada a sua situação:

```
 ----------------------------------------------
|        |         ARQUIVO        | DIRETÓRIO  |
| UMASK  |------------------------|            |
|        |   Binário   |  Texto   |            |
|------------------------------- -|------------|
|   0    |    r-x      |   rw-    |    rwx     |
|   1    |    r--      |   rw-    |    rw-     |
|   2    |    r-x      |   r--    |    r-x     |
```

```
|   3   |     r--      |   r--   |     r--      |
|   4   |     --x      |   -w-   |     -wx      |
|   5   |     ---      |   -w-   |     -w-      |
|   6   |     --x      |   ---   |     --x      |
|   7   |     ---      |   ---   |     ---      |
-------------------------------------------------
```

Um *arquivo texto* criado com o comando `umask 012;touch texto.txt` receberá as permissões `-rw-rw-r--`, pois 0 (dono) terá permissões `rw-`, 1 (grupo), terá permissões `rw-` e 2 (outros usuários) terão permissões `r--`. Um *arquivo binário* copiado com o comando `umask 012;cp /bin/ls /tmp/ls` receberá as permissões `-r-xr--r-x` (confira com a tabela acima).

Por este motivo é preciso atenção antes de escolher a umask, um valor mal escolhido poderia causar problemas de acesso a arquivos, diretórios ou programas não sendo executados. O valor padrão da umask na maioria das distribuições atuais é 022. A umask padrão no sistema Debian é a 022 .

A umask é de grande utilidade para programas que criam arquivos/diretórios temporários, desta forma pode-se bloquear o acesso de outros usuários desde a criação do arquivo, evitando recorrer ao `chmod`.

Capítulo 12

Redirecionamentos e Pipe

Esta seção explica o funcionamento dos recursos de direcionamento de entrada e saída do sistema `GNU/Linux`.

12.1 >

Redireciona a saída padrão de um programa/comando/script para algum dispositivo ou arquivo ao invés do dispositivo de saída padrão (tela). Quando é usado com arquivos, este redirecionamento cria ou substitui o conteúdo do arquivo.

Por exemplo, você pode usar o comando `ls` para listar arquivos e usar `ls >listagem` para enviar a saída do comando para o arquivo `listagem`. Use o comando `cat` para visualizar o conteúdo do arquivo `listagem`.

O mesmo comando pode ser redirecionado para o segundo console `/dev/tty2` usando: `ls >/dev/tty2`, o resultado do comando `ls` será mostrado no segundo console (pressione `ALT` e `F2` para mudar para o segundo console e `ALT` e `F1` para retornar ao primeiro). O mesmo resultado pode ser obtido com o comando `ls 1>/dev/tty2`, sendo que o número `1` indica que será capturada a saída padrão do comando.

Para redirecionar somente a saída de erros do comando `ls`, use a sintaxe: `ls 2>/tmp/erros-do-ls`

12.2 >>

Redireciona a saída padrão de um programa/comando/script para algum dispositivo ou adiciona as linhas ao final de arquivo ao invés do dispositivo de saída padrão (tela). A diferença entre este redirecionamento duplo e o simples, é se caso for usado com arquivos, adiciona a saída do comando ao final do arquivo existente ao invés de substituir seu conteúdo. .

Por exemplo, você pode acrescentar a saída do comando `ls` ao arquivo `listagem` do capítulo anterior usando `ls / >>listagem`. Use o comando `cat` para visualizar o conteúdo do arquivo `listagem`.

12.3 <

Direciona a entrada padrão de arquivo/dispositivo para um comando. Este comando faz o contrário do anterior, ele envia dados ao comando.

Você pode usar o comando `cat <teste.txt` para enviar o conteúdo do arquivo `teste.txt` ao comando `cat` que mostrará seu conteúdo (é claro que o mesmo resultado pode ser obtido com `cat teste.txt` mas este exemplo serviu para mostrar a funcionalidade do <).

12.4 <<

Este redirecionamento serve principalmente para marcar o fim de exibição de um bloco. Este é especialmente usado em conjunto com o comando `cat`, mas também tem outras aplicações. Por exemplo:

```
 cat << final
este arquivo
será mostrado
até que a palavra final seja
localizada no inicio da linha
final
```

12.5 | (pipe)

Envia a saída de um comando para a entrada do próximo comando para continuidade do processamento. Os dados enviados são processados pelo próximo comando que mostrará o resultado do processamento.

Por exemplo: `ls -la | more`, este comando faz a listagem longa de arquivos que é enviado ao comando `more` (que tem a função de efetuar uma pausa a cada 25 linhas do arquivo).

Outro exemplo é o comando `locate find | grep "bin/"`, neste comando todos os caminhos/arquivos que contém *find* na listagem serão mostrados (inclusive man pages, bibliotecas, etc.), então enviamos a saída deste comando para `grep "bin/"` para mostrar somente os diretórios que contém binários. Mesmo assim a listagem ocupe mais de uma tela, podemos acrescentar o `more`: `locate find | grep "bin/" | more`.

Podem ser usados mais de um comando de redirecionamento (<, >, |) em um mesmo comando.

12.6 Diferença entre o "|" e o ">"

A principal diferença entre o "|" e o ">", é que o Pipe envolve processamento entre comandos, ou seja, a saída de um comando é enviado a entrada do próximo e o ">" redireciona a saída de um comando para um arquivo/dispositivo.

Você pode notar pelo exemplo acima (`ls -la | more`) que ambos `ls` e `more` são comandos porque estão separados por um "|". Se um deles não existir ou estiver digitado incorretamente, será mostrada uma mensagem de erro.

Um resultado diferente seria obtido usando um ">" no lugar do "|"; A saída do comando `ls -la > more` seria gravada em um arquivo chamado `more`.

12.7 tee

Envia ao mesmo tempo o resultado do programa para a saída padrão (tela) e para um arquivo. Este comando deve ser usado com o pipe "|".

`comando |tee [arquivo]`

Exemplo: `ls -la | tee listagem.txt`, a saída do comando será mostrada normalmente na tela e ao mesmo tempo gravada no arquivo `listagem.txt`.

Capítulo 13

Impressão

Este capitulo descreve como imprimir em seu sistema GNU/Linux e as formas de impressão via spool, rede, gráfica, etc.

Antes de seguir os passos descritos neste capítulo, tenha certeza que seu kernel foi compilado com o suporte a impressora USB e/ou paralela ativado, caso contrário até mesmo a impressão direta para a porta de impressora falhará. .

13.1 Portas de impressora

Uma porta de impressora é o local do sistema usado para se comunicar com a impressora. Em sistemas GNU/Linux, a porta de impressora paralela é identificada como lp0, lp1, lp2 no diretório /dev, caso a impressora seja USB, o dispositivo será o mesmo, mas estará disponível no diretório /dev/usb. Os dispositivos lp0, lp1elp2 correspondem respectivamente a LPT1, LPT2 e LPT3 no DOS e Windows. Recomendo que o suporte a porta paralela esteja compilado como módulo no kernel.

13.2 Imprimindo diretamente para a porta de impressora

Isto é feito direcionando a saída ou o texto com > diretamente para a porta de impressora no diretório /dev.

Supondo que você quer imprimir o texto contido do arquivo trabalho.txt e a porta de impressora em seu sistema é /dev/usb/lp0, você pode usar os seguintes comandos:

- `cat trabalho.txt >/dev/usb/lp0` - Direciona a saída do comando cat para a impressora USB conectada em lp0.
- `cat <trabalho.txt >/dev/usb/lp0`. Faz a mesma coisa que o acima.
- `cat -n trabalho.txt >/dev/usb/lp0` - Numera as linhas durante a impressão.
- `head -n 30 trabalho.txt >/dev/usb/lp0` - Imprime as 30 linhas iniciais do arquivo.

- `cat trabalho.txt|tee /dev/usb/lp0` - Mostra o conteúdo do `cat` na tela e envia também para a impressora USB.

Os métodos acima servem somente para imprimir em modo texto (letras, números e caracteres semi-gráficos).

OBS: Note que a impressora somente imprimirá diretamente a partir da porta, caso ela seja uma impressora com firmware interna (impressora inteligente). Algumas impressoras mais recentes (principalmente os modelos mais baratos) somente imprimem caso estejam configuradas com o respectivo driver (Win Printers ou impressoras via software), e nunca aceitarão o comando diretamente para a porta de impressão. Para *Win Printers*, a melhor alternativa de configuração de funcionamento será através do CUPS (Common Unix Print System).

13.3 Imprimindo via spool

A impressão via spool (fila de impressão) tem por objetivo liberar logo o programa do serviço que está fazendo a impressão deixando um outro programa especifico tomar conta.

Este programa é chamado de *daemon de impressão*, normalmente é o `lpr` ou o `lprng` (recomendado) em sistemas `GNU/Linux`.

Logo após receber o arquivo que será impresso, o programa de spool gera um arquivo temporário (normalmente localizado em `/var/spool/lpd`) que será colocado em fila para a impressão (um trabalho será impresso após o outro, em seqüência). O arquivo temporário gerado pelo programa de spool é apagado logo após concluir a impressão.

Antes de se imprimir qualquer coisa usando os daemons de impressão, é preciso configurar os parâmetros de sua impressora no arquivo `/etc/printcap`. Um arquivo `/etc/printcap` para uma impressora local padrão se parece com o seguinte:

```
lp|Impressora compatível com Linux
 :lp=/dev/lp0
 :sd=/var/spool/lpd/lp
 :af=/var/log/lp-acct
 :lf=/var/log/lp-errs
 :pl#66
 :pw#80
 :pc#150
 :mx#0
 :sh
```

É possível também compartilhar a impressora para a impressão em sistemas remotos, isto será visto em uma seção separada neste guia.

Usando os exemplos anteriores da seção `Imprimindo diretamente para uma porta de impressora`, vamos acelerar as coisas:

- `cat trabalho.txt |lpr` - Direciona a saída do comando `cat` para o programa de spool `lpr`.

- `cat <trabalho.txt |lpr`. Faz a mesma coisa que o acima.
- `cat -n trabalho.txt |lpr` - Numera as linhas durante a impressão.
- `head -n 30 trabalho.txt |lpr` - Imprime as 30 linhas iniciais do arquivo.

A fila de impressão pode ser controlada com os comandos:
- `lpq` - Mostra os trabalhos de impressão atuais
- `lprm` - Remove um trabalho de impressão

Ou usado o programa de administração `lpc` para gerenciar a fila de impressão (veja a página de manual do `lpc` ou digite `?` ao iniciar o programa para detalhes).

13.4 Impressão em modo gráfico

A impressão em modo gráfico requer que conheça a marca e modelo de sua impressora e os métodos usados para imprimir seus documentos. Este guia abordará somente a segunda recomendação :-)

13.4.1 Ghost Script

O método mais usados pelos aplicativos do `GNU/Linux` para a impressão de gráficos do *Ghost Script*. O Ghost Script (chamado de *gs*) é um interpretador do formato *Pos Script* (arquivos `.ps`) e pode enviar o resultado de processamento tanto para a tela como impressora. Ele está disponível para diversas plataformas e sistema operacionais além do `GNU/Linux`, inclusive o `DOS`, `Windows`, `OS/2`, etc.

O formato `.ps` esta se tornando uma padronização para a impressão de gráficos em `GNU/Linux` devido a boa qualidade da impressão, liberdade de configuração, gerenciamento de impressão feito pelo *gs* e por ser um formato universal, compatíveis com outros sistemas operacionais.

Para imprimir um documento via Ghost Script, você precisará do pacote `gs`, `gsfonts` (para a distribuição `Debian` e distribuições baseadas, ou outros de acordo com sua distribuição Linux) e suas dependências. A distribuição `Debian` vem com vários exemplos Pos Script no diretório `/usr/share/doc/gs/example` que são úteis para o aprendizado e testes com o Ghost Script.

Hora da diversão:
- Copie os arquivos `tiger.ps.gz` e `alphabet.ps.gz` do diretório `/usr/share/doc/gs/examples` (sistemas `Debian`) para `/tmp` e descompacte-os com o comando `gzip -d tiger.ps.gz e gzip -d alphabet.ps.gz`. Se a sua distribuição não possui arquivos de exemplo ou você não encontra nenhuma referência de onde se localizam, mande um e-mail que os envio os 2 arquivos acima (são 32Kb).
- O Ghost Script requer um monitor EGA, VGA ou superior para a visualização dos seus arquivos (não tenho certeza se ele funciona com monitores CGA ou Hércules Monocromático) . Para visualizar os arquivos na tela digite:

  ```
  gs tiger.ps
  gs alphabet.ps
  ```

Para sair do `Ghost Script` pressione `CTRL+C`. Neste ponto você deve ter visto um desenho de um tigre e (talvez) letras do alfabeto. Se o comando `gs alphabet.ps` mostrou somente uma tela em branco, você se esqueceu de instalar as fontes do Ghost Script (estão localizadas no pacote `gsfonts` na distribuição Debian).

- Para imprimir o arquivo `alphabet.ps` use o comando:

  ```
  gs -q -dSAFER -dNOPAUSE -sDEVICE=epson -r240x72 -sPAPERSIZE=legal -sOut
  alphabet.ps
  ```

  O arquivo `alphabet.ps` deve ser impresso. Caso aparecerem mensagens como `Error: /invalidfont in findfont` no lugar das letras, você se esqueceu de instalar ou configurar as fontes do Ghost Script. Instale o pacote de fontes (`gsfonts` na `Debian`) ou verifique a documentação sobre como configurar as fontes. Cada uma das opções acima descrevem o seguinte:

  - `-q, -dQUIET` - Não mostra mensagens de inicialização do Ghost Script.
  - `-dSAFER` - É uma opção para ambientes seguros, pois desativa a operação de mudança de nome e deleção de arquivo e permite somente a abertura dos arquivos no modo somente leitura.
  - `-dNOPAUSE` - Desativa a pausa no final de cada página processada.
  - `-sDEVICE=dispositivo` - Dispositivo que receberá a saída do Ghost Script. Neste local pode ser especificada a marca o modelo de sua impressora ou um formato de arquivo diferente (como pcxmono, bmp256) para que o arquivo `.ps` seja convertido para o formato designado. Para detalhes sobre os dispositivos disponíveis em seu Ghost Script, digite `gs --help|less` ou veja a página de manual. Normalmente os nomes de impressoras e modelos são concatenados, por exemplo, bjc600 para a impressora *Canon BJC 600*, epson para impressoras padrão epson, stcolor para *Epson Stylus color*, etc. O Hardware-HOWTO contém referências sobre hardware suportados pelo `GNU/Linux`, tal como impressoras e sua leitura pode ser útil.
  - `-r<ResH>x<ResV>` - Define a resolução de impressão (em dpi) Horizontal e Vertical. Os valores dependem de sua impressora.
  - `-sPAPERSIZE=tamanho` - Tamanho do papel. Podem ser usados a4, legal, letter, etc. Veja a página de manual do gs para ver os outros tipos suportados e suas medidas.
  - `-sOutputFile=dispositivo` - Dispositivo que receberá a saída de processamento do gs. Você pode especificar
    - `arquivo.epson` - Nome do arquivo que receberá todo o resultado do processamento. O `arquivo.epson` terá toda a impressão codificada no formato entendido por impressoras epson e poderá ser impresso com o comando `cat arquivo.epson >/dev/lp0`. Uma curiosidade útil: É possível imprimir este arquivo em outros sistemas operacionais, tal como o `DOS` digitando: `copy /b arquivo.eps prn` (lembre-se que o `DOS` tem um limite de 8 letras no nome do arquivo e 3 na extensão. Você deve estar compreendendo a flexibilidade que o `GNU/Linux` e suas ferramentas permitem, isso é só o começo.
    - `impressao%d.epson` - Nome do arquivo que receberá o resultado do processamento. Cada página será gravada em arquivos separados como `impressao1.epson`, `impressao2.epson`. Os arquivos podem ser impressos usando os mesmos métodos acima.

* /dev/lp0 para uma impressora em /dev/lp0
* - para redirecionar a saída de processamento do gs para a saída padrão. É útil para usar o gs com pipes |.
* \|lpr - Envia a saída do Ghost Script para o daemon de impressão. O objetivo é deixar a impressão mais rápida.

Se você é curioso ou não esta satisfeito com as opções mostradas acima, veja a página de manual do gs.

13.5 Magic Filter

O *Magic Filter* é um filtro de impressão inteligente. Ele funciona acionado pelo spool de impressão (mais especificamente o arquivo /etc/printcap) e permite identificar e imprimir arquivos de diversos tipos diretamente através do comando lpr arquivo.

É um ótimo programa e **ALTAMENTE RECOMENDADO** se você deseja apenas clicar no botão imprimir e deixar os programas fazerem o resto :-) A intenção do programa é justamente automatizar os trabalhos de impressão e spool.

A maioria dos programas para ambiente gráfico X11, incluindo o Netscape, Word Perfect, Gimp e Star Office trabalham nativamente com o magicfilter.

13.5.1 Instalação e configuração do Magic Filter

O Magic Filter é encontrado no pacote magicfilter da distribuição Debian e baseadas.

Sua configuração pode ser feita com o programa magicfilterconfig que torna o processo de configuração rápido e fácil para quem não conhece a sintaxe do arquivo /etc/printcap ou não tem muitas exigências sobre a configuração detalhada da impressora.

Após instalar o magicfilter reinicie o daemon de impressão (se estiver usando a Debian, entre no diretório /etc/init.d e como usuário root digite ./lpr restart ou ./lprng restart).

Para testar o funcionamento do magicfilter, digite lpr alphabet.ps e lpr tiger.ps, os arquivos serão enviados para o magicfilter que identificará o arquivo como *Pos Script*, executará o Ghost Script e retornará o resultado do processamento para o daemon de impressão. O resultado será visto na impressora.

Se tiver problemas, verifique se a configuração feita com o magicfilterconfig está correta. Caso precise re-configurar o magicfilter, digite magicfilterconfig --force (lembre-se que a opção –force substitui qualquer configuração personalizada que tenha adicionado ao arquivo /etc/printcap).

13.5.2 Outros detalhes técnicos sobre o Magic Filter

Durante a configuração do magicfilter, a seguinte linha é adicionada ao arquivo /etc/printcap:

```
:if=/etc/magicfilter/epson9-filter
```

Não tenho nenhum contrato de divulgação com a *epson* :-) estou usando esta marca de impressora porque é a mais tradicional e facilmente encontrada. A linha que começa com `:if` no `magicfilter` identifica um arquivo de filtro de impressão.

O arquivo `/etc/magicfilter/epson9-filter` é criado usando o formato do magicfilter, e não é difícil entender seu conteúdo e fazer algumas modificações:

```
#! /usr/sbin/magicfilter
#
# Magic filter setup file for 9-pin Epson (or compatible) printers
#
# This file is in the public domain.
#
# This file has been automatically adapted to your system.
#
# wild guess: native control codes start with ESC
0       \033            cat

# PostScript
0 %! filter /usr/bin/gs -q -dSAFER -dNOPAUSE -r120x72 -sDEVICE=epson -sOutput
0 \004%! filter /usr/bin/gs -q -dSAFER -dNOPAUSE -r120x72 -sDEVICE=epson -sOu

# PDF
0 %PDF fpipe /usr/bin/gs -q -dSAFER -dNOPAUSE -r120x72 -sDEVICE=epson -sOutpu

# TeX DVI
0 \367\002 fpipe /usr/bin/dvips -X 120  -Y 72  -R -q -f

# compress'd data
0 \037\235 pipe /bin/gzip  -cdq

# packed, gzipped, frozen and SCO LZH data
0 \037\036 pipe /bin/gzip  -cdq
0 \037\213 pipe /bin/gzip  -cdq
0 \037\236 pipe /bin/gzip  -cdq
0 \037\240 pipe /bin/gzip  -cdq

0 BZh pipe /usr/bin/bzip2  -cdq

# troff documents
0 .\?\?\040 fpipe  `/usr/bin/grog  -Tps $FILE`
0 .\\\" fpipe  `/usr/bin/grog  -Tps $FILE`
0 '\\\" fpipe  `/usr/bin/grog  -Tps $FILE`
0 '.\\\" fpipe  `/usr/bin/grog  -Tps $FILE`
```

```
0  \\\" fpipe  `/usr/bin/grog  -Tps $FILE`
```

Você deve ter notado que para cada tipo de arquivo existe o respectivo programa que é executado, basta você modificar as opções usadas nos programas neste arquivo (como faria na linha de comando) para afetar o comportamento da impressão.

Por exemplo, modificando a resolução para -r240x72 no processamento de arquivos Pos Script (gs), a impressora passará a usar esta resolução.

Capítulo 14

X Window (ambiente gráfico)

Este capítulo do guia traz explicações sobre o ambiente gráfico X Window System.

14.1 O que é X Window?

É um sistema gráfico de janelas que roda em uma grande faixa de computadores, máquinas gráficas e diferentes tipos de máquinas e plataformas Unix. Pode tanto ser executado em máquinas locais como remotas através de conexão em rede.

14.2 A organização do ambiente gráfico X Window

Em geral o ambiente gráfico X Window é dividido da seguinte forma:

- `O Servidor X` - É o programa que controla a exibição dos gráficos na tela, mouse e teclado. Ele se comunica com os programas cliente através de diversos métodos de comunicação.

  O servidor X pode ser executado na mesma máquina que o programa cliente esta sendo executado de forma transparente ou através de uma máquina remota na rede.

- `O gerenciador de Janelas` - É o programa que controla a aparência da aplicação. Os gerenciadores de janelas (window managers) são programas que atuam entre o servidor X e a aplicação. Você pode alternar de um gerenciador para outro sem fechar seus aplicativos.

  Existem vários tipos de gerenciadores de janelas disponíveis no mercado entre os mais conhecidos posso citar o `Window Maker (feito por um Brasileiro)`, o `After Step`, `Gnome`, `KDE`, `twm` (este vem por padrão quando o servidor X é instalado), `Enlightenment`, `IceWm`, etc.

  A escolha do seu gerenciador de janelas é pessoal, depende muito do gosto de cada pessoa e dos recursos que deseja utilizar.

- `A aplicação cliente` - É o programa sendo executado.

Esta organização do ambiente gráfico X traz grandes vantagens de gerenciamento e recursos no ambiente gráfico UNIX, uma vez que tem estes recursos você pode executar seus programas em computadores remotos, mudar totalmente a aparência de um programa sem ter que fecha-lo (através da mudança do gerenciador de janelas), etc.

14.3 Iniciando o X

O sistema gráfico X pode ser iniciado de duas maneiras:

- `Automática` - Usando um gerenciador de seção como `xdm`, `gdm` ou `wdm` que apresenta uma tela pedindo nome e senha para entrar no sistema (login). Após entrar no sistema, o X executará um dos gerenciadores de janelas configurados.
- `Manual` - Através do comando `startx`, ou `xinit` . Neste caso o usuário deve entrar com seu nome e senha para entrar no modo texto e então executar um dos comandos acima. Após executar um dos comandos acima, o servidor X será iniciado e executará um dos gerenciadores de janelas configurados no sistema.

14.4 Servidor X

Como dito acima, o servidor X controla o teclado, mouse e a exibição dos gráficos em sua tela. Para ser executado, precisa ser configurado através do arquivo `/etc/X11/xorg.conf`, usando `dpkg-reconfigure xserver-xorg`, ou usando o utilitário `xf86cfg` (modo texto).

A finalização do servidor X é feita através do pressionamento simultâneo das teclas `CTRL`, `ALT`, `Back Space`. O servidor X é imediatamente terminado e todos os gerenciadores de janelas e programas clientes são fechados.

CUIDADO: Sempre utilize a opção de saída de seu gerenciador de janelas para encerrar normalmente uma seção X11 e salve os trabalhos que estiver fazendo antes de finalizar uma seção X11. A finalização do servidor X deve ser feita em caso de emergência quando não se sabe o que fazer para sair de um gerenciador de janelas ou de um programa mal comportado.

Recomendo fazer a leitura de 'Fechando um programa quando não se sabe como sair' on page 58 caso estiver em dúvidas de como finalizar um programa mal comportado ou que não sabe como sair.

Capítulo 15

Como obter ajuda no sistema

Dúvidas são comuns durante o uso do `GNU/Linux` e existem várias maneiras de se obter ajuda e encontrar a resposta para algum problema. O `GNU/Linux` é um sistema bem documentado, provavelmente tudo o que imaginar fazer ou aprender já esta disponível para leitura e aprendizado. Abaixo segue algumas formas úteis para encontrar a solução de sua dúvida, vale a pena conhece-las.

15.1 Páginas de Manual

As *páginas de manual* acompanham quase todos os programas `GNU/Linux`. Elas trazem uma descrição básica do comando/programa e detalhes sobre o funcionamento de opção. Uma página de manual é visualizada na forma de texto único com rolagem vertical. Também documenta parâmetros usados em alguns arquivos de configuração.

A utilização da página de manual é simples, digite:

`man [seção] [comando/arquivo]`

onde:

seção É a seção de manual que será aberta, se omitido, mostra a *primeira* seção sobre o comando encontrada (em ordem crescente).

comando/arquivo Comando/arquivo que deseja pesquisar.

A navegação dentro das páginas de manual é feita usando-se as teclas:

- q - Sai da página de manual
- PageDown ou f - Rola 25 linhas abaixo
- PageUP ou w - Rola 25 linhas acima
- SetaAcima ou k - Rola 1 linha acima
- SetaAbaixo ou e - Rola 1 linha abaixo
- r - Redesenha a tela (refresh)
- p ou g - Inicio da página
- h - Ajuda sobre as opções da página de manual
- s - Salva a página de manual em formato texto no arquivo especificado (por exemplo: `/tmp/ls`).

Exemplo, `man ls`, `man 5 hosts_access`.

15.2 Info Pages

Idêntico as páginas de manual, mas é usada navegação entre as páginas. Se pressionarmos <Enter> em cima de uma palavra destacada, a `info pages` nos levará a seção correspondente. A *info pages* é útil quando sabemos o nome do comando e queremos saber para o que ele serve. Também traz explicações detalhadas sobre uso, opções e comandos.

Para usar a info pages, digite:

`info [`*`comando/programa`*`]`

Se o nome do *comando/programa* não for digitado, a info pages mostra a lista de todos os manuais de *comandos/programas* disponíveis. A navegação da info pages é feita através de nomes marcados com um "*" (hipertextos) que se pressionarmos <Enter>, nos levará até a seção correspondente. A *info pages* possui algumas teclas de navegação úteis:

- q - Sai da info pages
- ? - Mostra a tela de ajuda (que contém a lista completa de teclas de navegação e muitos outras opções).
- n - Avança para a próxima página
- p - Volta uma página
- u - Sobre um nível do conteúdo (até checar ao índice de documentos)
- m - Permite usar a localização para encontrar uma página do `info`. Pressione `m`, digite o comando e tecle <Enter> que será levado automaticamente a página correspondente.
- d - Volta ao índice de documentos.

Existem muitos outras teclas de navegação úteis na info pages, mas estas são as mais usadas. Para mais detalhes, entre no programa `info` e pressione `?`.

Exemplo, `info cvs`.

15.3 Help on line

Ajuda rápida, é útil para sabermos quais opções podem ser usadas com o comando/programa. Quase todos os comandos/programas `GNU/Linux` oferecem este recurso que é útil para consultas rápidas (e quando não precisamos dos detalhes das páginas de manual). É útil quando se sabe o nome do programa mas deseja saber quais são as opções disponíveis e para o que cada uma serve. Para acionar o *help on line*, digite:

`[`*`comando`*`] --help`

comando - é o comando/programa que desejamos ter uma explicação rápida.

O Help on Line não funciona com comandos internos (embutidos no Bash), para ter uma ajuda rápida sobre os comandos internos, veja 'help' on the next page.

Por exemplo, `ls --help`.

15.4 help

Ajuda rápida, útil para saber que opções podem ser usadas com os *comandos internos* do interpretador de comandos. O comando `help` somente mostra a ajuda para comandos internos, para ter uma ajuda similar para comandos externos, veja 'Help on line' on the facing page. Para usar o `help` digite:

`help [comando]`

Por exemplo, `help echo`, `help exit`

15.5 apropos/whatis

Apropos procura por *programas/comandos* através da descrição. É útil quando precisamos fazer alguma coisa mas não sabemos qual comando usar. Ele faz sua pesquisa nas páginas de manual existentes no sistema e lista os comandos/programas que atendem a consulta. Para usar o comando `apropos` digite:

`apropos [descrição]`

Digitando `apropos copy`, será mostrado todos os comandos que tem a palavra `copy` em sua descrição (provavelmente os programas que copiam arquivos, mas podem ser mostrados outros também).

15.6 locate

Localiza uma palavra na estrutura de arquivos/diretórios do sistema. É útil quando queremos localizar onde um comando ou programa se encontra (para copia-lo, curiosidade, etc). A pesquisa é feita em um banco de dados construído com o comando `updatedb` sendo feita a partir do diretório raíz `/` e sub-diretórios. Para fazer uma consulta com o `locate` usamos:

`locate [expressão]`

A *expressão* deve ser o nome de um arquivo diretório ou ambos que serão procurados na estrutura de diretórios do sistema. Como a consulta por um programa costuma localizar também sua página de manual, é recomendável usar *"pipes"* para filtrar a saída do comando (para detalhes veja '| (pipe)' on page 106 .

Por exemplo, para listar os diretórios que contém o nome *"cp"*: `locate cp`. Agora mostrar somente arquivos binários, usamos: `locate cp|grep bin/`

15.7 which

Localiza um programa na estrutura de diretórios do path. É muito semelhante ao `locate`, mas a busca é feita no `path` do sistema e somente são mostrados arquivos executáveis .

`which [programa/comando].`

15.8 Documentos HOWTO's

São documentos em formato *texto*, *html*, etc, que explicam como fazer determinada tarefa ou como um programa funciona. Normalmente são feitos na linguagem `SGML` e convertidos para outros formatos (como o texto, HTML, Pos Script) depois de prontos.

Estes trazem explicações detalhadas desde como usar o `bash` até sobre como funciona o modem ou como montar um *servidor internet completo*. Os HOWTO´s podem ser encontrados no diretório do projeto de documentação do `GNU/Linux` (LDP) em ftp://metalab.unc.edu/pub/Linux/docs/HOWTO/ ou traduzidos para o Português pelo LDP-BR em http://www.tldp.org/projetos/howto/traduzidos.php. Caso tenha optado por instalar o pacote de HOWTO's de sua distribuição `GNU/Linux`, eles podem ser encontrados em: `/usr/doc/how-to`

15.9 Documentação de Programas

São documentos instalados junto com os programas. Alguns programas também trazem o *aviso de copyright, changelogs, modelos, scripts, exemplos e FAQs (perguntas freqüêntes)* junto com a documentação normal.

Seu princípio é o mesmo do How-to; documentar o programa. Estes arquivos estão localizados em:

`/usr/share/doc/[programa].`

Programa é o nome do programa ou comando procurado.

15.10 FAQ

FAQ é um arquivo de perguntas e respostas mais freqüêntes sobre o programa. Normalmente os arquivos de FAQ estão localizados junto com a documentação principal do programa em `/usr/share/doc/[programa].`

15.11 Internet

Certamente o melhor suporte ao `GNU/Linux` é via Internet, veja abaixo alguns locais úteis de onde pode obter ajuda ou se atualizar.

15.11.1 Páginas Internet de Referência

Existem boas páginas Internet Nacionais e Internacionais sobre o GNU/Linux e assuntos relacionados com este sistema. A maioria trazem documentos e explicações sobre configuração, instalação, manutenção, documentação, suporte, etc.

Estas páginas podem ser encontradas através de ferramentas de busca. Entre outras páginas, posso citar as seguintes:

- http://www.debianbrasil.org/ Projeto Debian-Br. A Debian é uma distribuição de Linux conhecida por sua qualidade, grande número de pacotes, estabilidade, facilidade de atualização, desenvolvimento aberto, segurança, ferramentas de gerenciamento de servidores e comprometimento com o software livre.

  A Debian é feita originalmente em inglês e traduzida por grupos em vários lugares do mundo. O projeto **Debian-br** destina-se a colaborar na tradução da Debian para o Português (nossa língua-mãe). Através desse projeto, todos poderão, da forma colaborativa como na Debian, trazer essa excelente distribuição em nosso idioma!

  Participe:
  - Você pode pegar um documento pra traduzir
  - Reformular a página do projeto
  - Programando para o projeto
  - Sendo um desenvolvedor da Debian
  - A pagina do projeto é a http://www.debianbrasil.org/
  - Revisar documentação
  - Ou participar de outras tarefas do seu interesse!

  Entre em contato com o responsável pelo projeto pelo email <debian-br@listas.cipsga.org.br> para saber como entrar no projeto ou visite a página http://www.debianbrasil.org/. Todos os interessados estão convidados a participar do projeto!

- http://www.br-linux.org/ - Boletim diário com as noticias mais recentes sobre GNU/Linux, testes, redes, descrição/configuração/ avaliação de programas, entrevistas, downloads, dica do dia, mecanismo de busca no site, links, etc. Em Português.

  Responsável pela página: Augusto Campos <brain@matrix.com.br> endereço: http://www.br-linux.org/.

- http://www.olinux.com.br/ - Trata o GNU/Linux com o foco jornalístico e tem a intenção de prover informações eficazes e esclarecedoras capazes de instruir, reciclar e tornar acessível aos usuários o conhecimento e aprofundamento de temas relacionados a plataforma GNU/Linux.

  Publicação diária de Artigos que são feitos para que o usuário possa resolver problemas e tirar dúvidas deste sistema. Assuntos diversos sobre programas, serviços e utilitários. Também conta com seções de programação, jogos, segurança e entrevistas com personalidades do cenário *software livre/código aberto*. Atualização diária.

  Responsável pela página: Linux Solutions <baptista@linuxsolutions.com.br> endereço: http://www.olinux.com.br/.

- http://www.linuxsecurity.com.br/ - Boletins de segurança, publicações de textos nacionais, traduções de sites especializados em segurança, programas relacionados com criptografia e segurança no ambiente Linux. A página requer um navegador com suporte a Java.

  Endereço: http://www.linuxsecurity.com.br/.

- http://www.tldp.org/ - Projeto de documentação do GNU/Linux no Brasil. Toda a documentação traduzida para o Português do Brasil pode ser encontrada lá.

  Responsável pela página: <ricardo@conectiva.com.br> endereço: http://www.tldp.org/.

- http://www.noticiaslinux.com.br/ -

  Notícias diárias sobre GNU/Linux e Software Livre no site, por e-mail ou RSS.

  Responsável pela página: <deivison@noticiaslinux.com.br> endereço: http://www.noticiaslinux.com.br/

- http://www.linux.org/ - Página oficial do GNU/Linux mantida pela *Transmeta* (a empresa que Linus Torvalds vem trabalhando atualmente). Muita referência sobre GNU/Linux, distribuições, hardwares, softwares, downloads, etc.

  Responsável pela página: <webmaster@linux.org> endereço: http://www.linux.org/.

- http://counter.li.org/ - Este é um serviço que tem o objetivo de contar os usuários, máquinas, grupos de usuários Linux existentes ao redor do mundo. Te encorajo a se registrar neste site e indica-lo aos seus amigos, é de **graça**, você estará contribuindo para o aumento das estatísticas do número de usuários no mundo, país, sua cidade, etc.

  O site também conta com um sistema de estatísticas de usuários, máquinas e grupos de usuários espalhados ao redor do mundo. Você pode saber em poucos segundos a quantidade de usuários Linux em seu país, cidade, etc.

  Responsável pela página: Harald T. Alvestrand <harald@alvestrand.no> endereço: http://counter.li.org/.

- http://metalab.unc.edu/ - O ponto de referência mais tradicional de softwares GNU/Linux do mundo. Você pode encontrar desde dicas, documentação (todos os How-Tos) até diversas distribuições GNU/Linux.

  Responsável pelo site: <webmaster@sunsite.unc.edu> endereço: http://metalab.unc.edu/.

- http://www.themes.org/ - Neste site você encontra milhares de temas divididos em categorias para os mais diversos gerenciadores de janelas no GNU/Linux. O site é muito pesado, por causa das fotos, é recomendável um bom fax-modem ou muita paciência.

  Responsável pela página: <webmaster@themes.org> endereço: http://www.themes.org/.

- http://www.oreill.com/safari/ - Neste site você encontra os livros publicados sobre a licença OpenBook da Orreil. Na maioria livros que não atende mais propósitos atualmente e livros em que os autores concordaram em licenciar sob os termos OpenBook.

  Endereço: http://www.oreill.com/safari/.

Caso conhecer uma página de Internet que contenha materiais úteis a comunidade GNU/Linux ou desejar incluir a sua, entre em contato para sua inclusão na próxima versão do guia junto com uma descrição da página.

15.11.2 Listas de discussão

São grupos de usuários que trocam mensagens entre si, resolvem dúvidas, ajudam na configuração de programas, instalação, etc. É considerado o melhor suporte ao GNU/Linux pois qualquer participante pode ser beneficiar das soluções discutidas. Existem milhares de listas de discussões sobre o GNU/Linux espalhadas pelo mundo, em Português existem algumas dezenas.

Algumas listas são específicas a um determinado assunto do sistema, algumas são feitas para usuários iniciantes ou avançados, outras falam praticamente de tudo. Existem desde usuários iniciantes, hackers, consultores, administradores de redes experientes e gurus participando de listas e oferecendo suporte de graça a quem se aventurar em instalar e usar o sistema GNU/Linux.

A lista de discussão funciona da seguinte forma: você se inscreve na lista enviando uma mensagem ao endereço de inscrição, será enviada um pedido de confirmação por e-mail, simplesmente dê um reply na mensagem para ser cadastrado. Pronto! agora você estará participando do grupo de usuários e receberá todas as mensagens dos participantes do grupo. Assim você poderá enviar sua mensagem e ela será vista por todos os participantes da lista.

Da mesma forma, você pode responder uma dúvida de outro usuário da lista ou discutir algum assunto, tirar alguma dúvida sobre a dúvida de outra pessoa, etc.

Não tenha vergonha de enviar sua pergunta, participar de listas de discussão é uma experiência quase obrigatório de um Linuxer. Abaixo segue uma relação de listas de discussão em Português com a descrição, endereço de inscrição, e o que você deve fazer para ser cadastrado:

<debian-user-portuguese@lists.debian.org> Lista de discussão para usuários Portugueses da Debian. Também são discutidos assuntos relacionados ao Linux em geral. A inscrição é aberta a todos os interessados.

Para se inscrever, envie uma mensagem para <debian-user-portuguese-request@lists.debian.org> contendo a palavra subscribe no assunto da mensagem. Será enviada uma mensagem a você pedindo a confirmação da inscrição na lista de discussão, simplesmente dê um reply na mensagem (responder) e você estará cadastrado e poderá enviar e receber mensagens dos participantes.

<debian-news-portuguese@lists.debian.org> A Debian é extremamente bem estruturada quanto a divulgações e notícias, várias listas de email e várias páginas compõe essa base. A *Debian Weekly News* é especialmente importante pois dá uma visão geral do que se passou na Debian durante a semana. E não traz apenas traduções mas também adições dos acontecimentos atuais da Debian no Brasil, ou projetos concluídos ou lançados pela equipe *Debian-br* (http://www.debianbrasil.org/).

Essa lista NÃO é usada para resolução de dúvidas e problemas, apenas para o RECEBIMENTO de notícias relacionadas a Debian. Não poste mensagens nela!

Para se inscrever, envie uma mensagem para <debian-news-portuguese-request@lists.debian.org> contendo a palavra subscribe no assunto da mensagem. Será enviada uma mensagem a você pedindo a confirmação da inscrição na lista de discussão, simplesmente dê um reply na mensagem (responder) e você passará a receber as notícias sobre a Debian em Português.

<linux-br@unicamp.br> Lista de discussão que cobre assuntos diversos. Esta lista é voltada para usuários com bons conhecimentos no GNU/Linux, são abordados assuntos como redes, configurações, etc. Esta é uma lista moderada, o que significa que a mensagem que envia passam por uma pessoa que verifica (modera) e a libera caso estejam dentro das normas adotada na lista. É uma lista de alto nível e recomendada para quem deseja fugir de mensagens como não consigo instalar o Linux, não sei compilar o kernel, o que eu faço quando vejo uma tela com o nome login:?, etc.

Para se inscrever nesta lista, envie uma mensagem para: <linux-br-request@unicamp.br> contendo a palavra subscribe no assunto da mensagem e aguarde o recebimento da confirmação da inscrição. Apenas responda a mensagem de confirmação para se inscrever. Para se descadastrar envie uma mensagem para o mesmo endereço mas use a palavra unsubscribe.

<dicas-l@unicamp.br> Esta lista envia diariamente uma dica de Unix, sistemas da Microsoft ou novidades da Internet.

Para se inscreve nesta lista de discussão, envie uma mensagem para: <dicas-l-request@unicamp.br> contendo a palavra subscribe no corpo da mensagem e aguarde o recebimento da confirmação da inscrição. Apenas responda a mensagem de confirmação para confirmar sua inscrição na lista. Para se descadastrar envie uma mensagem para o mesmo endereço mas use a palavra unsubscribe.

Esta listagem deveria estar mais completa, mas eu não lembro de todas as listas!. Também recomendo dar uma olhada em 'Listas de Discussão via Email' on page 129 que descreve recomendações de comportamento em listas de discussão.

15.12 Netiqueta

São recomendações que tem como objetivo facilitar a para comunicação através dos recursos de uma rede. O nome *Netiqueta* vem de "Etiqueta de Rede" (*Net Etiquete*). O material desta seção

foi escrito com base nos anos de observação que tive via internet e também com referência a rfc 1855.

15.12.1 Recomendações Gerais sobre a Comunicação Eletrônica

- Como recomendação geral, lembre-se que a conversa via internet é feita sempre de uma para outra pessoa ou de uma para várias pessoas, e que a forma de comunicação é a mesma que utilizaria se estivesse de frente a frente com a pessoa. Nunca diga algo que não diria se estivesse diante da outra pessoa. Existem pessoas que por estar atrás de um monitor, se sentem "maiores" se esquecendo disso e causando prejuízos de comunicação (e sem imaginar que a pessoa do outro lado da linha existe). Apesar do modo que as frases são escritas expressarem o jeito que a outra pessoa está do outro lado da linha e seu tom de comunicação no decorrer da conversar, existem algumas coisas que não podem ser totalmente expressadas através da Internet, como por exemplo a expressão da "face" das pessoas. Para isto foram criados símbolos chamados *smileys* que expressam a face da outra pessoa em determinado momento, e dependendo do sentido da conversa, um smiley pode expressar corretamente a intenção de sua frase. Os mais usados são os seguintes:

  ```
  :-)   --> Sorriso

  :-(   --> Triste

  ;-)   --> Piscadinha

  :-O   --> De boca aberta

  :-|   --> Sem graça

  8-)   --> De óculos

  |-)   --> Com sono e feliz

  <:-)  --> Bobo
  ```

  Para entender o sentido do smiley, veja ele de lado (45 graus). Use os smileys em suas conversas, mas com cautela. Não espere que a inclusão de um smiley sorridente ":-)" deixe o destinatário da mensagem contente com um comentário rude ou insulto.
- ESCREVER EM MAIÚSCULAS significa gritar quando escrever mensagens eletrônicas.
- Use *asteriscos* para destacar uma palavra ou frase. _Isso_ indica uma palavra/frase sublinhada.
- Se você troca mensagens com pessoas do mundo todo, não espere que um japonês responda logo seu e-mail que enviou as 15:00 da tarde. A essa hora no país dele, ele está roncando forte na cama e sonhando com a placa 3D que vai ganhar para melhorar o desempenho de seus jogos de `Linux`.
- Durante a comunicação com pessoas de diferentes regiões (ou países), evite a utilização de gírias, ou expressões regionais. Uma interpretação em uma determinada região não

garante que ela tenha o mesmo significado para seu destinatário, as vezes pode ser até ofensiva.

- Assuma que sua mensagem está trafegando sobre uma via não segura, desta forma não envie informações pessoais que não enviaria em uma carta comum. O uso de criptografia pode garantir melhor segurança na transmissão de dados.

15.12.2 Email

- Tenha o hábito de colocar sempre um assunto na mensagem que envia para identificar seu conteúdo.
- Respeite os direitos autorais das mensagens de e-mail. Se precisar encaminhar mensagens, preserve seu conteúdo original.
- Procure limitar o tamanho da linha a 70 caracteres. Muitos usuários utilizam cliente de e-mail em modo texto, e nem todo mundo usa a mesma resolução que você.
- Caso o e-mail que responda tenha mais que 100 linhas, é recomendável colocar a palavra "LONGA" no assunto da mensagem. Se possível corte as partes não necessárias da mensagens de respostas tendo o cuidado de não "cortar" de forma mal educada a mensagem de outra pessoa.
- Caso utiliza um editor programa de e-mails com suporte a HTML, envie o e-mail utilizando ambos os formatos TEXTO e HTML, muitos administradores Linux utilizam sistemas que não suportam HTML.
- Não espere que o espaçamento ou desenhos ASCII usados em uma mensagem sejam mostrados corretamente em todos os sistemas.
- Utilize sempre uma assinatura no final da mensagem para identificar você e principalmente seu endereço de e-mail. Em alguns cliente de e-mail, o campo `Reply-to` é bagunçado, e em e-mails redirecionados o endereço de resposta é excluído. A assinatura facilita encontrar o remetente da mensagem. Tente manter a assinatura em um tamanho de no máximo 4 linhas.
- Não repasse mensagens de corrente por e-mail. Elas tem somente o objetivo de espalhar boatos na Internet e se espalhar. Normalmente elas vem com uma história bonita e no final diz se não repassar acontecerá tudo ao contrário com você ou algo do tipo. *Não vai acontecer nada! ignore isso e não entre na corrente!* Pelas políticas da Internet, você pode ter sua conta de e-mail perdida se fizer mal uso dele.

15.12.3 ICQ/MSN/Gtalk/Skype

Ferramentas de mensagens instantâneas são eficientes, alertando a presença on-line do usuário, auxiliando na redução de custos, etc. Este documento inclui algumas recomendações etiqueta para melhor os usuários aproveitarem melhor ferramentas de comunicaçao que seguem o padrão IM:

- De atenção ao status da outra pessoa. Se ela estiver "on-line" ou "free for chat" significa que ela está desocupada e que pode conversar naquele instante. Se estiver como não perturbe, envie somente mensagens se for mesmo preciso.
- EVITE colocar nicks chamativos e caracteres exóticos. Nem todos os usuários vêem o nick da mesma forma que a pessoa que os colocou.

- Seja também sensato ao usar ferramentas de mensagem instantanea. Não entre nele caso não possa conversar, ou avise isso mudando seu status para o mais adequado para a situação, assim os outros poderão entender que está longe do computador, não disponível ou ocupado.
- É recomendável ser prudente quanto ao envio de mensagens, não envie mais do que 4 mensagens seguidas, pois a outra pessoa terá dificuldades para responder a todas elas mais outra que talvez possa estar recebendo de outras (ou nem tenha recebido, caso exista algum problema temporário no servidor).
- Guarde seu login e senha em lugar seguro. Caso ela seja perdida, você terá trabalho para avisar a todos de sua lista de contato.
- Sempre que enviar uma URL, procure do que se trata na mensagem.
- No modo de chat, use as recomendações descritas sobre o talk (em 'Talk' on the current page).
- Como em toda comunicação on-line, seja cauteloso quando a pessoa que conversa. Nem sempre quem conversamos do outro lado é a pessoa que esperamos encontrar. Lembre-se que um registro falso e uma identidade pode ser criada sem dificuldades por qualquer pessoa.

15.12.4 Talk

- Use sempre quebra de linhas ao escrever suas mensagens, use pelo menos 70 caracteres para escrever suas mensagens de talk. Evita escrever continuamente até a borda para fazer quebra de linha automática, alguns clientes de talk não aceitam isso corretamente.
- Sempre que termina uma frase, deixe uma linha em branco (tecle enter 2 vezes) para indicar que a outra pessoa pode iniciar a digitação.
- Sempre se despeça da outra pessoa e espere ela responder antes de fechar uma seção de conversação. O respeito mútuo durante um diálogo é essencial :-)
- Lembre-se que o talk normalmente interrompe as pessoas que trabalham nativamente no console. Evite dar talk para estranhos, pois podem fazer uma má impressão de você. Tente antes estabelecer outros meios de comunicação.
- Se a outra pessoa não responder, não assuma de cara que ela está ignorando você ou não levando sua conversa muito bem. Ela pode simplesmente estar ocupada, trabalhando, ou com problemas no cliente de talk. Alguns cliente de talk dão problemas durante a comunicação remota, lembre-se também que sua comunicação é via UDP :-)
- Se a pessoa não responder seus talks durante certo tempo, não deixe ele infinitamente beepando a pessoa. Tente mais tarde :-)
- Seja atencioso caso utilize mais de uma seção de talk ao mesmo tempo.
- O talk também leva em consideração sua habilidade de digitação. Muitos erros e correções contínuas fazem a outra pessoa ter uma noção de você, suas experiências, etc ;-)

15.12.5 Listas de Discussão via Email

- Tente se manter dentro do assunto quando responder mensagens de listas. Seja claro e explicativo ao mesmo tempo :-)
- Sempre coloque um assunto (subject) na mensagem. O assunto serve como um resumo

do problema ou dúvida que tem. Alguns usuários, principalmente os que participam de várias listas de discussão, verificam o assunto da mensagem e podem simplesmente descartar a mensagem sem lê-la porque as vezes ele não conhece sobre aquele assunto.

- Nunca use "Socorro!", "Help!" ou coisa do gênero como assunto, seja objetivo sobre o problema/dúvida que tem: *"Falha ao carregar módulo no do kernel"*, *"SMAIL retorna a mensagem Access denied"*, *"Novidades: Nova versão do guia Foca Linux"* ;-).
- Procure enviar mensagens em formato `texto` ao invés de `HTML` para as listas de discussão pois isto faz com que a mensagem seja vista por todos os participantes (muitos dos usuários `GNU/Linux` usam leitores de e-mail que não suportam formato html) e diminui drásticamente o tamanho da mensagem porque o formato texto não usa tags e outros elementos que a linguagem HTML contém (muitos dos usuários costumam participar de várias listas de discussão, e mensagens em HTML levam a um excesso de tráfego e tempo de conexão).
- Tenha cautela e bom censo em suas mensagens para listas e grupos de discussão, considere que cada mensagem que posta é são arquivadas para futura referência.
- Quando o conteúdo das mensagem tomar outro rumo, é ético modificar o assunto do e-mail para se adequar ao novo conteúdo da mensagem. Por exemplo, `Correção nas regras de Netiqueta` para `Conversa de pessoa para pessoa (Era: Correção das regras de Netiqueta)`.
- Quando a conversa em grupo sair do assunto e envolver apenas duas pessoas, é conveniente retirar os endereços das pessoas/listas do CC.
- Não mande arquivos grandes para as listas, principalmente se eles tiverem mais que 40Kb de tamanho. Se precisar enviar arquivos maiores que isso, envie diretamente para os e-mails dos interessados depois de perguntar.
- Quando enviar mensagens para listas de discussão, seja educado e cordial quanto ao conteúdo de sua mensagem. Envie CC's para as pessoas que dizem respeito ao assunto, assim com a lista.
- Tente ignorar ou não responda mensagens de "Guerras" em listas (*Flame Wars*), caso queira reponde-la por algum tipo de agressão de quem mandou a mensagem, esperar para responde-la a noite (nunca é garantida uma boa resposta no momento que está de cabeça quente). Lembre-se de quando responde uma mensagem de "Flame War" a "altura" de quem mandou seus ataques, está sendo igualmente tão baixo quando o "nível" dessa pessoa.
- Caso se desentenda com alguma pessoa em uma lista de discussão, não envie mensagens agressivas para a listas, se precisar, faça isso diretamente para a pessoa! Você pode se arrepender disso mais tarde.
- Não culpe o administrador da lista pelos usuários que participam dela. Notifique somente usuários que não estejam colaborando com a lista e outras coisas que prejudiquem seu funcionamento. Administradores preservam o funcionamento das listas, e não o comportamento dos usuários.
- Não use auto respostas para listas de discussão. Pelos inconvenientes causados, você pode ser descadastrado ou banido de se inscrever na lista/newsgroup.
- Salve as mensagens de inscrição que recebe da lista. Ela contém detalhes sobre seus recursos, e a senha usada muitas vezes para se descadastrar dela ou modificar suas permissões de usuário. O administrador pode te ajudar nessa tarefa, mas não espere que ele esteja sempre disponível para realizar tarefas que podem ser feitas pelo próprio usuário.

- Muitas pessoas reclamam do excesso de mensagens recebidas das listas de discussão. Se você recebe muitas mensagens, procure usar os *filtros de mensagens* para organiza-las. O que eles fazem é procurar por campos na mensagem, como o remetente, e enviar para um local separado. No final da filtragem, todas as mensagens de listas de discussão estarão em locais separados e as mensagens enviadas diretamente a você entrarão na caixa de correio principal, por exemplo. Um filtro de mensagens muito usado no `GNU/Linux` é o `procmail`, para maiores detalhes consulte a documentação deste programa. O Netscape também tem recursos de filtros de mensagem que podem ser criadas facilmente através da opção "Arquivo/Nova SubPasta" ("File/New Subfolder") do programa de E-mail. Então defina as regras através do menu "Editar/Filtros de Mensagens" ("Edit/Message filters") clicando no botão "Novo"("New").

Capítulo 16

Apêndice

Este capítulo contém considerações sobre o guia Foca GNU/Linux.

16.1 Sobre este guia

O guia Foca foi criado em 12 de Novembro de 1999. A versão que está lendo tem o objetivo de servir como referência a usuários *Iniciantes* e que estão tendo o primeiro contato com o sistema operacional `GNU/Linux`, ou com referência de consulta rápida. Uma versão que abrange um diferente nível de aprendizado ou mais completa pode ser baixada de Página Oficial do guia Foca GNU/Linux (http://www.guiafoca.org).

A versão que esta lendo agora foi gerada com as seguintes opções:

- Descrição detalhada de comandos
- Opções usadas em comandos e programas
- Observações sobre comandos e configurações
- Exemplos para a melhor compreensão do assunto discutido.

e contém o(s) nível(is) de aprendizado (Iniciante, Intermediário e Avançado):

- Iniciante

O *Foca GNU/Linux* é atualizado frequentemente, por este motivo recomendo que assine um dos canais de comunicação para ser informado de novas versões:

- Assinar o Twitter do focalinux <@focalinux>
- Assinar o RSS presente na página do guia e atualizações.
- A ficha do aviso de atualizações na página web em Página Oficial do guia Foca GNU/Linux (http://www.guiafoca.org) no fim da página principal. Após preencher a ficha do aviso de atualizações, eu te enviarei um e-mail sobre o lançamento de novas versões do guia e o que foi modificado, desta forma você poderá decidir em copia-la caso a nova versão contém modificações que considera importantes.

Versões diferentes deste guia podem ser geradas a partir do código fonte SGML ou obtidas através da home page principal (para detalhes veja 'Onde encontrar a versão mais nova do guia?' on page 135).

16.2 Sobre o Autor

Gleydson Mazioli da Silva é Capixaba, nascido em Vila Velha. Amante de eletrônica desde criança, foi atraido para a informática através da curiosidade em funcionamento e reparo de hardware.

Se dedica ao sistema `Linux` desde 1997. determinado na realização de testes de ferramentas e sistemas avaliando pontos fortes e fracos de cada uma. Logo que iniciou em `Linux` passou a estudar exaustivamente aspectos técnicos de distribuições e rede em `Linux`/`BSD`.

Entre coisas que gosta de fazer/implementar em `Linux`: possibilidade de pesquisa e atualização de conhecimento constante, automatização e tomada inteligente de decisões, níveis de segurança da informação (tanto físico e lógico), firewalls, virtualização, redes virtuais, integração de sistemas, forense computacional, documentação de processos, desenvolvimento de ferramentas GPL para a comunidade, depuração, desenvolvimento de documentações, etc.

Um dos desenvolvedores da distribuição *Liberdade*, *CAETECT*, *Debian-BR* e desenvolvedor oficial da distribuição *Debian*. Atuou como tradutor do LDP-BR, traduzindo vários HOW-TOs importantes para a comunidade Linux Brasileira. É um dos administradores do projeto CIPSGA, cuidando de uma infinidade de serviços que o projeto oferece a comunidade que deseja estrutura para hospedar, fortalecer e manter projetos em software livre.

Trabalhou para algumas empresas do Espírito Santo, no Governo Federal e de estados na implantação de sistemas em software livre. Atualmente atua como gerente de tecnologia da Spirit Linux, uma empresa focada na inovação, avaliação e integração de ferramentas de código aberto junto a seus clientes.

Concorda com certificações, mas destaca que o mais importante é aproveitar a oportunidade dada pela certificação para estudo e auto avaliação de seus pontos fracos e assim procurar melhora-los. Possui certificação LPI nível 3 e um ISO9001 internacional em Administração Linux, como primeiro no ranking Brasileiro.

E-mail: E-mail: `<gleydson@guiafoca.org>`, Twitter: `<@gleydsonmazioli>`.

16.3 Referências de auxílio ao desenvolvimento do guia

- As seções sobre comandos/programas foram construídas após uso, teste e observação do comportamento das opções dos comandos/programas, help on line, páginas de manual, info pages e documentação técnica do sistema.
- How-tos do Linux (principalmente o *Networking Howto*, *Security-Howto*) ajudaram a formar a base de desenvolvimento do guia e desenvolver algumas seções (versões *Intermediário* e *Avançado* somente).
- Todos os exemplos e seções descritivas do guia são de minha autoria.
- Manual de Instalação da *Debian GNU/Linux* - Os capítulos contendo materiais extraídos do manual de instalação da Debian são muito úteis e explicativos, seria desnecessário reescrever um material como este. O texto é claro e didaticamente organizado, o documento aborda detalhes técnicos úteis sobre hardwares em geral e o Linux ausentes nos manuais de outras distribuições Linux.

16.4 Onde encontrar a versão mais nova do guia?

Novas versões deste guia, avisos de lançamento, outros níveis de aprendizado (Iniciante, Intermediário e Avançado), versões para outras distribuições Linux podem ser encontradas em: Página Oficial do guia Foca GNU/Linux (http://www.guiafoca.org).

Se quiser receber notificações de novas versões, use uma das formas a seguir:

- por E-Mail: envie uma mensagem para <gleydson@guiafoca.org> pedindo para ser incluído na lista de atualizações do guia ou preencha o formulário encontrado no final da Home Page do guia.
- Twitter: Assine o Twitter do guia Foca: @focalinux
- RSS: Assine o RSS na página oficial do guia (citado acima) para receber atualizações e novidades.

16.5 Colaboradores do Guia

Entre as principais colaborações até a versão atual, posso citar as seguintes:

- Djalma Valois <djalma@cipsga.org.br> - Pela atual hospedagem do Foca GNU/Linux. Estou muito feliz vendo o Foca GNU/Linux fazendo parte de um projeto tão positivo como o CIPSGA é para o crescimento e desenvolvimento do software livre nacional.

- Bakurih <bakurih@yahoo.com> - Revisão inicial do guia, após suas primeiras versões.

- Eduardo Marcel Maçan <macan@debian.org> - Pela antiga hospedagem, na época do site metainfo.

- Michelle Ribeiro <michelle@cipsga.org.br> - Por dispensar parte de seu atencioso tempo enviando revisões e sugestões que estão melhorando bastante a qualidade do guia. Entre eles detalhes que passaram despercebidos durante muito tempo no guia e página principal.

  E também por cuidar do fonte do guia ;-)

- Augusto Campos <brain@matrix.com.br> - Descrição sobre a distribuição Suse em 'Distribuições do Linux' on page 7 .

- Paulo Henrique Baptista de Oliveira <baptista@linuxsolutions.com.br> - Pelo apoio moral oferecido durante os freqüentes lançamentos do guia, acompanhamento e divulgação.

- Diego Abadan <diego@hipernet.ufsc.br> - Envio de correções significativas, novos endereços de listas de discussão.

- Alexandre Costa <alebyte@bol.com.br> - Envio de centenas de patches ortográficos nas versões Iniciante e Intermediário do guia que passaram desapercebidas durante várias versões do guia...
- Christoph Simon <ciccio@prestonet.com.br> - Pela pesquisa e a gigantesca coletânea de textos sobre o Linux enviada. Eles estão sendo muito úteis tanto para mim quanto no desenvolvimento do guia.
- Gustavo Noronha <dockov@zaz.com.br> - Vem enviando freqüentes correções, contribuições construtivas ao desenvolvimento além de apoio ao desenvolvimento do guia . Vale a pena destaca-lo por sua atual dedicação junto a distribuição Debian/GNU, sua tradução e a comunidade Open Source.
- Pedro Zorzenon Neto <pzn@debian.org> - Envio de diversas atualizações para o nível Avançado, principalmente sobre o firewall iptables.

16.6 Marcas Registradas

Todas as marcas registradas citadas neste guia são propriedades de seus respectivos autores.

16.7 Futuras versões

Estes são os materiais que pretendo adicionar em futuras versões do guia:

- Acrescentar mais detalhes sobre o sistema gráfico X-Window.
- Entre outros ítens que venho estudando para verificar se encaixam no perfil do guia.

Esta é uma futura implementação que venho estudando para acompanhar o crescimento do guia. Sugestões são bem vindas e podem ser enviadas para <gleydson@guiafoca.org>.

16.8 Chave Pública PGP

Chaves PGP são usadas para criptografar arquivos, e-mails ou qualquer outra coisa que desejamos que somente uma pessoa tenha acesso. O PGP segue o padrão de chave pública/privada; a chave pública é distribuída a todos e a chave privada permanece na posse do criador para que ele seja o único a ter acesso aos dados criptografados após digitar a "frase de acesso" correta.

Minha chave PGP segue abaixo, ela também pode ser encontrada em http://pgp.ai.mit.edu. Se você deseja saber mais sobre o PGP, recomendo um excelente documento encontrado na seção Apostilas em http://www.cipsga.org.br/

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v1.0.6 (GNU/Linux)
Comment: For info see http://www.gnupg.org

mQGiBDl7WYgRBACsQNtIozvf8XId+xEpF2D1x7nqgFdJyn1QA2VzXg0/OZ9DewXj
qr7ChEIoyyzAmxBSubE/jdtkAb9+2LsE9+OXgzJvBc4luYpv+HG2IXlMPujI9drO
ubLlK6xqPiakBgqBTS74rp/ZEEAGQsr0sug7b8nsXHMk+spyGkjsU8pPWwCgltai
4vfmBDMZMqBYvUoksVxbaKcD/ApAMghgE53KAAKFtwXI0o7K1DJmdZBufCvGDbEB
Y3MVS4BI+aXxoP5zQpEmQ5+lYOZ8RjPL9pNUJa9nOQtjf7Kiw/41BPDtlZXCeRR5
OcQTit0lYRCLGam7FZ22uliwh0h/3lpf4olMff3qeLqv1DECbo8Qsdn6yxynLihE
OA9kA/9K1sqiIl/+gXM3/Sjz8EcrwQNklV3MoaETbDmukbXcOEUjdqfFr1xARM5W
8SKoVrWO5y1oa1e9XcQuK6g8c7KeJsK/GEWYiRwX2X2AqdBC2ZzVfJSmgpguZJHn
ltMdYZhPwZaCsNPdQSlem3UrGupL0pbpT7PqkvyAHBH2itB9X7RKR2xleWRzb24g
TWF6aW9saSBkYSBTaWx2YSAoQ2hhdmUgUEdQIFBlc3NvYWwpIDxnbGV5ZHNvbkBl
c2NlbHNhbmV0LmNvbS5icj6IVgQTEQIAFgUCOXtZiAQLCgQDAxUDAgMWAgECF4AA
CgkQpWvD35hbooFdwgCfQijPTW5VH+Cep1HIBvyuw9uMg7wAoI/RYW0tkjjnhrgH
8+Zqx6AgGlQ/iEYEEBECAAYFAjnlrPAACgkQoUSye+uc2tWZPgCfVgR4lbd8XPBm
bjPupLzB3EYAPI8AoJomkfsgz+NuUZy1mD6pI1Ptc/fDiEYEEBECAAYFAjm4FfUA
CgkQco65AkzGCoF34gCgsVcH4b3s6kfCtjD7iMMhkubnDnUAoL2UiorB3Z/m3f9A
RZiRMhQUclMRiEYEEBECAAYFAjm4ITAACgkQt1anjIgqbEupXgCg1/NjvT562Hgt
/ft5JETOf3yOFywAn1SmK3unyhMU5GU9d49MNM3fNgBtiEYEEBECAAYFAjnFWrYA
CgkQORwuc54x+1t8VQCeMZTCla98rrI60EnlkAvb9AaScm4AnA4V795vcVlr3ix9
f6fcl5YGamKciEYEEBECAAYFAjvSF6sACgkQUZATEoypqPVQ7wCbBTRiSGGMzMTd
KJotfRKf5aoUAr0AoIAX0oE5XEEFm7Ea0IQqG91T9TvXtDtHbGV5ZHNvbiBNYXpp
b2xpIGRhIFNpbHZhIChEZXZlbG9wZXIpIDxnbGV5ZHNvbkBkZWJpYW4ub3JnPohX
BBMRAgAXBQI7BR7fBQsHCgMEAxUDAgMWAgECF4AACgkQpWvD35hbooESRACcCliY
yxR02KEBYs8cxKav9L0wlzwAn2Z9DWAbqi9Mv4fqPqZ7mViSMRbeiEYEEBECAAYF
AjsauX0ACgkQt1anjIgqbEvBEACffJxYfK22YPQ8ZkcjIc85BCiPLuUAnRq1EE9i
ukdUHPUo0vzHBeiN355miEYEEBECAAYFAjxEY28ACgkQGERS+iaKCE2fgwCeNGNV
Mpa1EWgXF+Hj15gidVjaVCAAn187X6eATJAVzspveNSf/Ny1iuFnuQENBDl7WasQ
BACxhBiSFOGa8tv7MOn0XVa6WCViBuQs9QJx2ZnMrx/KssRHMsNXnps+i+zVENqr
1Lz5zPpP7eWgrUy6B7/V9R4LV8nwHC1lZrR/1xyJ6G5j9RLSbYInZCLIAFUMlAar
iTThMhvXM+Pf7SXPj+ivrP9EYPSLxqTs1K/dWAbrDK/QiwADBQP9Hgc3EOw+7luB
/bXWssQp70bF9yvZLCGOgIE/rZIbOXumXkPlV7FTDgv+h47Bgcj2KDPEM98LUyxG
GcJAmrC9gWH7mYEUFNn1bGD+qHRwJ7+xj45NXBJDOBbHzTDS8QhacCRGW1CvRVgP
8ycPDOv/hmGfAJEzqzUkSO1uBcPmmXSIRgQYEQIABgUCOXtZqwAKCRCla8PfmFui
gQHnAJ4kDKHKvG9s9OjGV6RvszTDGE51igCcCZn0rO/Si0ek97bTCIusQzJF/pA=
=bvnT
-----END PGP PUBLIC KEY BLOCK-----
```